



Universitat Politècnica de Catalunya
Facultat d'Informàtica de Barcelona

Treball de Final de Grau

Manipulació de so digital amb DSP

Autor: Martí Roig Planas

Director: Antoni Grau Saldes

Departament: ESAII

Titulació: Grau en Enginyeria Informàtica

Curs: Q1 2016-17

Resum

La música ha estat present al llarg de la història de la humanitat, però no va ser fins als anys setanta que la producció musical va sofrir un avenç espectacular gràcies a la tecnologia. Amb la possibilitat de registrar sons, manipular-los i reproduir-los va néixer una nova era musical, i pocs anys més tard esdevenia l'era digital facilitant encara més la possibilitat de produir música.

Els sintetitzadors són eines que ajuden als productors de música a aconseguir una gran varietat de sons i faciliten la composició de peces musicals de qualitat. N'hi ha de molts tipus: seqüenciadors, teclats musicals, caixes de ritmes, samplers, etc...

En aquest projecte de final de grau es desenvolupa un seqüenciador de notes musicals digital, amb l'ajuda de microcontroladors vistos en assignatures del grau, de manera que el facin un aparell de baix consum útil per a un productor de música.

Aquest seqüenciador pot seqüenciar d'una a vuit notes. Cada nota pot tenir una durada d'un nivell a vuit nivells i es poden escollir tots els tons de la tercera i quarta octava de l'escala musical. Es poden guardar fins a set patrons de seqüenciació de notes. Disposa d'efectes de so. També disposa d'una pantalla LCD que mostra tota la informació de la configuració del sistema en temps real.

Per finalitzar, s'ha adaptat una caixa per posar-hi dins tot el sistema i només es vegi els elements d'interacció: els pulsadors, els potenciòmetres i la pantalla.

Resumen

La música ha sido presente a lo largo de la historia de la humanidad, pero no fue hasta finales de los años setenta que la producción musical sufrió un espectacular avance gracias a la tecnología. Con la posibilidad de registrar sonidos, manipularlos y reproducirlos nació una nueva era musical, y pocos años más tarde acontecía la era digital facilitando aún más la posibilidad de producir música.

Los sintetizadores son herramientas que ayudan a los productores de música a alcanzar una gran variedad de sonidos y facilitan la composición de piezas musicales de calidad. Existen de muchos tipos: secuenciadores, teclados musicales, cajas de ritmos, samplers...

En este proyecto de fin de grado se desarrolla un secuenciador de notas musicales digital, con la ayuda de microcontroladores vistos en asignaturas del grado, de modo que lo hace un dispositivo de bajo consumo útil para un productor musical.

Este secuenciador puede secuenciar de una a ocho notas. Cada nota puede tener una duración de un nivel a ocho niveles y se pueden escoger todos los tonos de la tercera y cuarta octava de la escala musical. Se pueden guardar hasta siete patrones de secuenciación de notas. Dispone de efectos de sonido. También dispone de una pantalla LCD que muestra toda la información de la configuración del sistema en tiempo real.

Para finalizar, se ha adaptado una caja para poner dentro todo el sistema y solo se vean los elementos de interacción: los pulsadores, los potenciómetros y la pantalla.

Abstract

Music has been present throughout the history of mankind, but it was not until the end of the seventies that musical production experienced a spectacular advance due to technology. With the possibility of recording sounds, manipulating and reproducing them, a new music era was born, and a few years later the digital world emerged easing the possibility of producing music.

Synthesizers are tools that help music producers achieve a wide variety of sounds and facilitate the composition of quality musical pieces. There are many types of synthesizers: sequencers, musical keyboards, drum machines, samplers...

In this final degree project a digital sequencer is developed, with the help of microcontrollers seen in grade subjects, making it a useful low consumption device for a music producer.

This sequencer can sequence from one to eight notes. Each note can have a duration of one level to eight levels and you can choose all the tones of the third and fourth octave of the musical scale. Up to seven sequencing patterns can be saved. It has sound effects. It also has an LCD screen that shows all the information of the system configuration in real time.

To complete, a box has been adapted to put the whole system inside and only the elements of interaction are seen: the pushbuttons, the potentiometers and the screen.

Índex

1	Introducció i contextualització.....	4
1.1	Contextualització.....	4
1.2	Actors implicats.....	5
1.3	Estat de l'art.....	5
1.3.1	Estudi de mercat.....	5
1.3.2	Conclusions.....	7
1.4	Formulació del problema.....	8
2	Abast del projecte.....	9
2.1	Abast.....	9
2.2	Possibles obstacles.....	10
2.2.1	Hardware.....	10
2.2.2	Software.....	13
2.3	Metodologia i rigor.....	13
2.3.1	Mètode de treball.....	13
2.3.2	Eines de seguiment.....	14
2.3.3	Mètode de validació.....	14
3	Planificació temporal.....	15
3.1	Durada estimada del projecte.....	15
3.2	Descripció de les tasques.....	15
3.2.1	Fita inicial: GEP.....	15
3.2.2	Compra de components.....	15
3.2.3	Configuració de l'entorn de treball.....	15
3.2.4	Desenvolupament del dispositiu.....	15
3.2.4.1	Programació de l'Arduino.....	16
3.2.4.2	Programació del DSP.....	16
3.2.4.3	Comunicació Arduino amb el DSP.....	17
3.2.5	Construcció de la caixa.....	17
3.2.6	Proves i verificació.....	18
3.2.7	Fita final.....	18
3.3	Recursos humans.....	19
3.4	Diagrama de Gantt.....	20
4	Gestió econòmica.....	21
4.1	Identificació i estimació dels costos.....	21
4.1.1	Costos directes.....	21
4.1.1.1	Costos en recursos humans.....	21
4.1.1.2	Costos Hardware.....	22
4.1.1.3	Costos Software.....	23
4.1.2	Costos indirectes.....	23
4.1.3	Imprevistos.....	24

4.1.4	Contingència i cost total.....	24
4.2	Control de gestió.....	25
4.3	Cost Final.....	25
4.4	Viabilitat comercial del producte.....	25
5	Sostenibilitat i compromís social.....	26
5.1	Dimensió ambiental.....	26
5.2	Dimensió econòmica.....	26
5.3	Dimensió social.....	27
5.4	Matriu de sostenibilitat.....	27
6	Fonaments teòrics.....	28
6.1	So.....	28
6.1.1	Propietats bàsiques.....	28
6.1.1.1	Freqüència.....	28
6.1.1.2	Amplitud.....	30
6.2	Àudio digital.....	31
6.2.1	Mostreig.....	31
6.2.1.1	Freqüència de mostreig.....	31
6.2.1.2	Resolució.....	32
6.2.2	Conversió analògic – digital.....	32
6.2.3	Conversió digital – analògic.....	33
6.2.4	Soroll.....	34
7	Elecció i descripció dels components.....	35
7.1	Arduino Mega 2560.....	35
7.2	TMS320C5515 eZdsp USB Stick.....	37
7.2.1	Processador TMS320C5515.....	37
7.2.2	TLV320AIC3204.....	38
7.2.3	Slot d'expansió.....	39
7.3	Pantalla LCD 16x2.....	40
7.4	Teclat WaveShare AD.....	41
7.5	Potenciòmetres.....	42
7.6	HUB USB.....	42
7.7	Caixa Retex.....	43
8	Descripció i configuració del software.....	44
8.1	Arduino.....	44
8.1.1	Pantalla.....	45
8.1.2	Polsadors.....	48
8.1.3	Potenciòmetres.....	48
8.1.4	I2C (Wire.h).....	49
8.1.5	EEPROM.....	51
8.2	C5515 eZdsp.....	52
8.2.1	Configuració del C5515.....	52
8.2.2	Estructura.....	54

8.2.3	Generació d'àudio.....	55
8.2.4	Efectes de so.....	56
8.2.4.1	Eco.....	56
8.2.4.2	Reverberació.....	57
8.2.5	I2C.....	58
8.2.6	Esriptura del programa a la memòria NOR.....	59
9	Disseny i construcció de la caixa.....	60
9.1	Disseny.....	60
9.2	Resultat final de la caixa.....	62
10	Treball futur.....	63
11	Conclusions.....	65
12	Bibliografia.....	66
13	Annexe.....	68
13.1	Índex de taules.....	68
13.2	Índex d'il·lustracions.....	69
13.3	Glossari d'abreviatures.....	70
13.4	Construcció de la caixa.....	72
13.5	Manual d'usuari del seqüenciador.....	76
13.5.1	Encesa.....	76
13.5.2	Potenciòmetres.....	77
13.5.3	Teclats.....	77

1 Introducció i contextualització

1.1 Contextualització

L'objectiu d'aquest projecte és crear un instrument musical digital: un seqüenciador de notes musicals amb funcionalitats de manipulació del so i amb la possibilitat de guardar patrons de notes.

Als anys 70 van aparèixer els sintetitzadors analògics, un instrument musical que genera ones elèctriques que es poden convertir en so mitjançant un amplificador i per tant generar melodies. La seva invenció va suposar una revolució per a la música, tant com per la influència en els gèneres i estils que ja existien tant com els que van néixer a partir d'aleshores, com la música electrònica.

Amb l'arribada del món digital i els microprocessadors s'ha pogut traslladar la funció dels sintetitzadors analògics a sintetitzadors digitals, fent-los un dispositiu més barat de construir, amb possibilitats d'efectes més amplis i sense perdre qualitat de so. També han aparegut els DAW (digital station workstation) que inclouen VSTi (virtual station technology instrument) de manera que els instruments són virtuals i no físics. Els VSTi tenen el gran avantatge que pots automatitzar els paràmetres virtualment de forma exacte i evidentment que no ocupen un espai físic, tot i així la majoria de productors els hi resulta molt més còmode tenir teclats i potenciòmetres reals per poder interactuar amb els instruments d'una forma més viva, és a dir que no tot soni tan exacte i perfecte, a part és més còmode a l'hora de cercar sons i experimentar amb l'instrument.

Existeixen molts models de sintetitzadors digitals, amb característiques molt diverses, amb mòduls addicionals d'efectes de so, amb i sense teclats, o simplement amb polsadors i potenciòmetres. Totes aquestes característiques els fan un producte competitiu i relativament costós econòmicament. L'idea d'aquest producte és aprofitar la senzillesa d'un Arduino i fer-lo funcionar juntament amb un DSP i crear un seqüenciador senzill però eficaç.

Per tant el producte final és un instrument destinat a productors musicals que busquen crear nous sons i melodies d'una forma senzilla, i amb baix cost econòmic.

1.2 Actors implicats

L'instrument està pensat per un ús personal i poder compondre produccions musicals pròpies, atès que no dispo de cap sintetitzador. Per tant la persona que utilitzarà el producte i se'n beneficiarà és el creador. Si el producte resultant dóna resultats millors dels esperats es pot estudiar dissenyar una placa PCB per a la seva producció i comercialització en massa.

El tutor d'aquest projecte és el professor Antoni Grau Saldes, encarregat de guiar i supervisar la realització del projecte controlant que es compleixin els objectius establerts.

1.3 Estat de l'art

1.3.1 Estudi de mercat

Part dels sintetitzadors més coneguts tenen incorporat un seqüenciador de notes o també anomenat *arpegiador*, s'ha cercat al mercat actual aquests sintetitzadors amb arpegiador i s'han analitzat les seves característiques.

Roland Jupiter-8[1]:

Aquest sintetitzador analògic del 1981 és dels més complets que hi ha, i a banda de l'arpegiador, té polifonia de 8 veus, 2 timbres, 2 VCOs per veu, 1 LFO de forma triangle/quadrada/serra/aleatori, 1 filtre passa-baixos, 1 equip ADSR per VCF, 1 equip ADSR per VCA, 64 patrons d'emmagatzematge i un teclat de 61 tecles. Al ser vintage i no fabricar-se està considerat de col·lecció i el seu preu en botigues online de segona mà pot arribar als 8000€ (eBay). El seu predecessor en el món digital és el model Jupiter-80 de característiques similars amb certes millores i amb el preu de 2700€



Il·lustració 1.1: Roland Jupiter-8

Roland Juno-60[2]:

Aquest és un altre sintetitzador analògic mític de la marca Roland llençat l'any 1982. A banda de l'arpegiador, té polifonia de 6 veus, 1 timbre, 1 DCO per veu (pols/serra/quadrat), 1 LFO de forma triangle, 1 equip ADSR, 1 filtre passa-baixos, 56 patrons d'emmagatzematge, efecte de chorus i un teclat de 61 teclcs. Com l'anterior és vintage i el seu preu de segona mà pot arribar als 1500€ (eBay).



Il·lustració 1.2: Roland Juno-60

Elektron Monomachine[3]:

Canviant al món digital, trobem la Elektron Monomachine que és un sintetitzador amb 6 veus, 6 timbres, 3 bancs de 6 oscil·ladors, 18 LFO's, 6 bancs de 128 patrons d'emmagatzematge i una gran varietat d'efectes com eco i reverberació. El seu preu és de 900€.



Il·lustració 1.3: Elektron Monomachine

Cyclone Analogic Bass Bot TT-303[4]:

Es tracta d'un sintetitzador analògic que imita el model Roland TB-303, considerat de col·lecció i per tant d'alt valor econòmic (2300€). Consisteix en un arpegiador bastant complet ja que té varis paràmetres configurables. El so que genera té un to 'àcid' que és molt peculiar, i famós, en el món de la música electrònica. El seu preu de mercat és de 425€.



Il·lustració 1.4: Cyclone Analogic TT-303 BassBot

1.3.2 Conclusions

Veient l'estat del mercat i l'evolució d'aquest, es veu una manca de la creació de sintetitzadors de baix cost econòmic i baix cost energètic. A més a més, molts cops els productors musicals només utilitzen una o dues funcionalitats de les moltes que pot oferir un sintetitzador, cosa que fa que estiguin invertint més diners dels desitjats, ja que com podem comprovar el cost dels sintetitzadors són molt elevats. Aquests sintetitzadors contenen moltes funcionalitats i tenen molts més elements d'interacció, de forma que necessiten energia externa, és a dir, estar connectats a la corrent. En canvi, pel producte desenvolupat en aquest projecte n'hi haurà prou en disposar d'una alimentació per USB. Això el farà un producte mòbil, per exemple podrà ser utilitzat mentre es viatge en tren o avió si es disposa d'un ordinador portàtil que l'alimenti. Per tant, el producte que es dissenyarà amb el projecte pot omplir el buit que hi ha en els sintetitzadors de baix cost econòmic i energètic, i a més, ser mòbil.

1.4 Formulació del problema

El projecte té com a principal objectiu desenvolupar un seqüenciador de notes de 8 passos amb manipulació de so, com és l'eco i la reverberació, i la possibilitat de guardar els patrons de notes desitjats a la memòria interna del dispositiu. Es pretén que la seva interacció sigui mitjançant pulsadors i potenciómetres i que una pantalla LCD mostri informació a l'usuari. El dispositiu el componen dos microcontroladors, un per a la interacció amb l'usuari i l'altre per a la generació de so. S'investigarà si amb un adaptador USB serà suficient per alimentar el dispositiu de baix consum. També es provarà de crear una caixa contenidora on posar el dispositiu de forma que només es vegin els elements d'interacció (pulsadors, potenciómetres i pantalla).

2 Abast del projecte

2.1 Abast

L'objectiu general del projecte és crear un seqüenciador per generar notes musicals i melodies, que inclogui manipulació de so i emmagatzemament de patrons de notes.

A partir d'aquí es pot especificar més en cada aspecte:

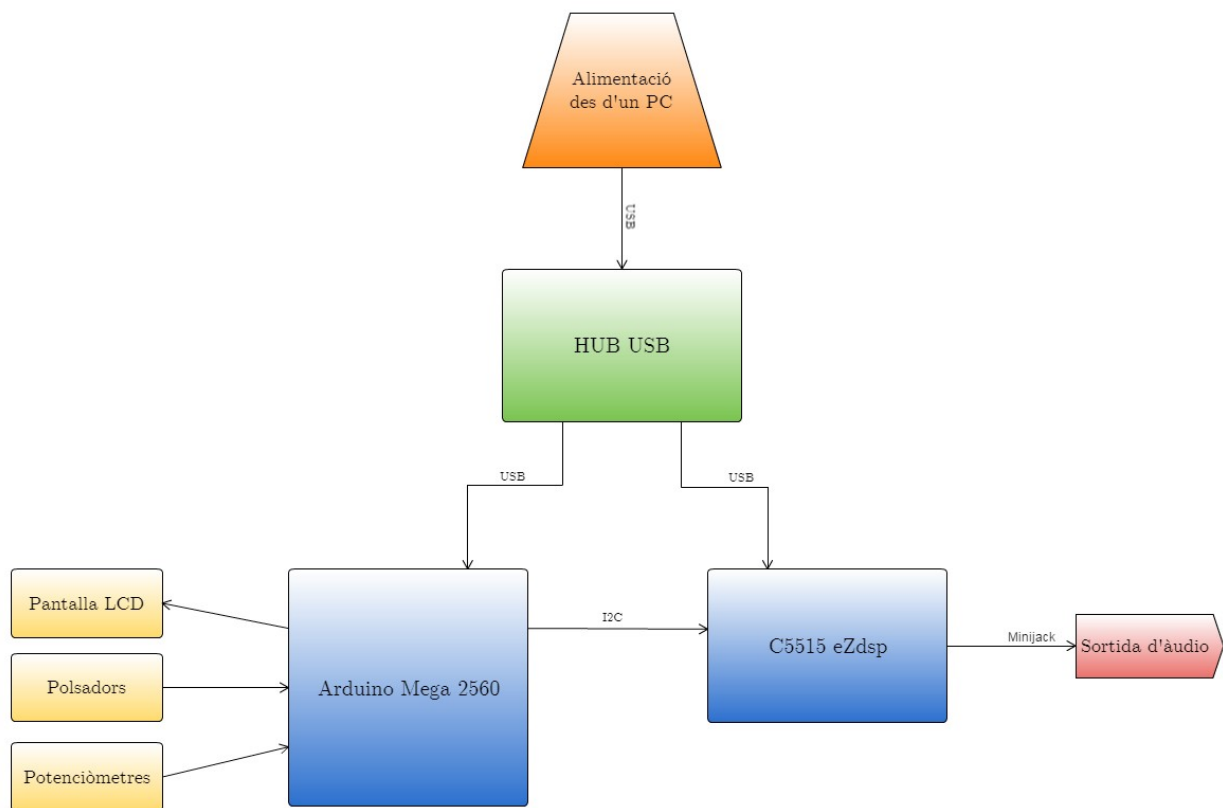
- Seqüenciació de 8 passos (o menys, configurable), 8 notes que es reproduïxen en bucle
- Manipulació de so amb efectes d'eco i de reverberació amb DSP, s'investigarà si és factible introduir altres efectes.
- Control de volum de l'àudio i del tempo de la seqüenciació
- Emmagatzemament de 8 patrons guardats a la memòria interna de l'Arduino
- Interacció amb el dispositiu mitjançant polsadors i potenciòmetres per poder controlar les característiques anteriors

2.2 Possibles obstacles

2.2.1 Hardware

Tant l'Arduino Mega com el C5515 eZdsp són microprocessadors de baix consum i una connexió USB des d'un ordinador hauria de ser suficient per alimentar tot el sistema. Per tal de connectar-los es fa servir un hub USB amb interruptor per si es vol encendre el circuit o apagar. L'Arduino llegeix dades dels pulsadors i dels potenciómetres, i envia dades a la pantalla LCD i dades al DSP. El DSP processa l'àudio i disposa d'una sortida minijack per connectar a un altaveu extern i poder escoltar el so.

L'esquema provisional del sistema és el següent:



Il·lustració 2.1: Disseny del sistema

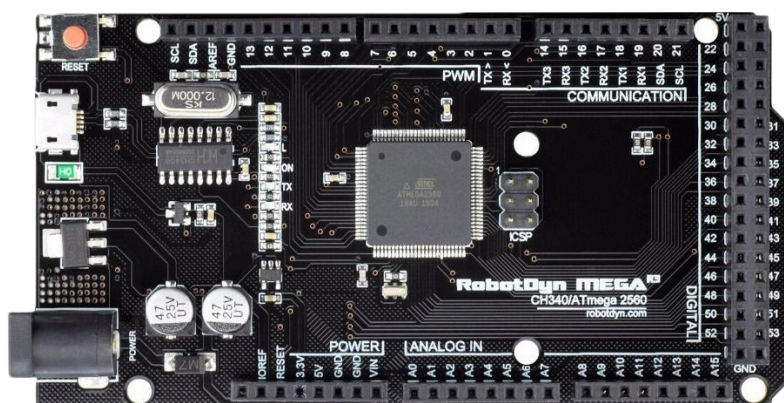
Arduino Mega 2560[5]:

S'ha triat aquest model de Arduino atès que disposa de 16 pins d'entrada analògica, això és de gran utilitat per poder tenir múltiples polsadors i potenciòmetres. Problemes que poden sortir a partir d'aquí són els rebots de lectura que apareixen al polsar un polsador. S'ha d'idear un codi de lectura de polsadors per tal d'evitar-ho.

Per tal de reduir costos del projecte s'ha triat un model clon de l'Arduino amb l'única diferència respecte l'original és que es connecta, i s'alimenta, per microUSB.



Il·lustració 2.2: Arduino Mega 2560 Original



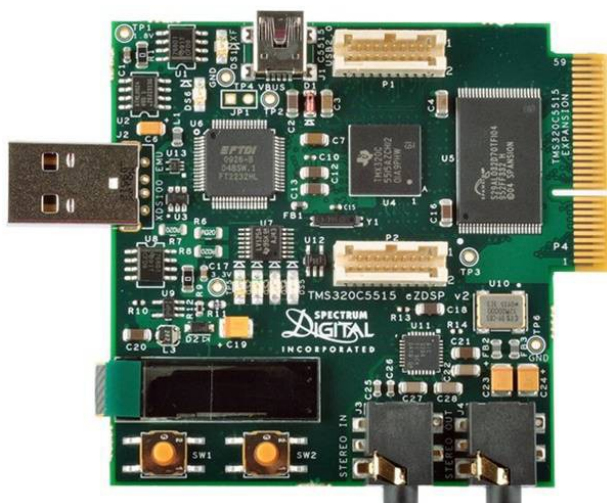
Il·lustració 2.3: Arduino Mega 2560 Clone

TMS320C5515 eZdsp USB Stick[6]:

Aquesta placa és l'encarregada de generar el so i també de generar els efectes d'aquest. El seu fabricant és Texas Instrument i ja ha estat utilitzada prèviament a una assignatura del grau. Algunes d'aquestes experiències prèvies han estat problemàtiques ja que s'usa un software proporcionat per Texas Instrument anomenat Code Composer Studio que pot arribar a ser inestable, i la programació i boot de la placa es fa feixuga i costosa.

Per processar l'àudio utilitza el còdec AIC3204, que permet l'entrada i sortida d'àudio de la placa, però només farem servir la sortida.

Un altre problema és que els pins del slot d'expansió són molt petits i estan molt a prop entre sí cosa que fa difícil soldar-hi cables, ja que es pot soldar més d'un pin per error. Es pot optar per utilitzar materials adhesius, tot i ser menys resistent. Almenys dos d'aquests pins seran necessaris per a fer la comunicació amb l'Arduino.



Il·lustració 2.4: TMS320C5515 eZdsp USB Stick

2.2.2 Software

Per a la programació d'Arduino s'utilitza el programa que proporciona el fabricant que és el Arduino 1.6.12[7], un programa open-source. Està disponible per Windows, OSX i Linux. Respecte la lectura de polsadors i potenciòmetres no haurien de suposar cap problema, excepte els rebots. La configuració de la pantalla tampoc hauria de ser un problema atès que hi ha recursos suficients a la xarxa.

Per a la programació del DSP s'utilitza el Code Composer Studio[8] proporcionat pel fabricant, que usa el framework de Eclipse, i que és gratuït per a estudiants. Només està disponible per Windows. Per tant el sistema operatiu que s'utilitzarà serà Windows.

El desenvolupament de software aleshores consistirà en dos programes, un per l'Arduino i l'altre pel DSP. A mida que es vagin desenvolupant els programes, es farà servir l'emmagatzemament al núvol mitjançant el servei Google Drive[9] per tenir còpies de seguretat.

2.3 Metodologia i rigor

2.3.1 Mètode de treball

Aquest projecte es pot dividir en 7 etapes:

1. Compra de components
2. Lectura polsadors i potenciòmetres
3. Programació pantalla LCD
4. Configuració comunicació Arduino-DSP
5. Programació seqüenciació DSP
6. Programació efectes de so DSP
7. Interconnexió dels components
8. Construcció d'una caixa

La compra de components és el primer pas del projecte, es compra en diferents webs online triant les més econòmiques però sense que el temps d'enviament sigui molt elevat ja que si no alenteix el desenvolupament del projecte.

Com les etapes 2, 3, 4, 5, 6 i 7 són paral·leles no suposaria un alentiment en el desenvolupament rebre components en diferents dates, ja que es podria treballar en la etapa que té els components disponibles mentre no han arribat altres components necessaris per altres etapes.

En un principi, les etapes anteriors esmentades es realitzaran de forma paral·lela utilitzant la metodologia àgil Scrum[10]. Segons la metodologia Scrum, un projecte es divideix en blocs, on cada bloc ha de tenir un resultat ràpid i complet. D'aquesta forma es poden intercalar objectius de les diferents etapes i anar progressant proporcionalment.

Un cop acabades les anteriors etapes es fabricaria la caixa contenidora per poder fer el dispositiu més fàcil de traslladar i robust.

2.3.2 Eines de seguiment

Com ja s'ha esmentat, es realitzaran còpies de seguretat del codi font al núvol. Es farà d'una forma ordenada i apuntant notes de l'estat del projecte en cada pujada. D'aquesta manera si sorgeixen problemes d'error en el codi es podrà tornar a una versió estable.

Amb el director del projecte s'aniran programant reunions per mostrar l'avanç del projecte i resoldre o redirigir els problemes que sorgeixin.

2.3.3 Mètode de validació

Per validar el treball desenvolupat, es carregaran els programes a l'Arduino i al DSP al finalitzar un bloc o objectiu menor per tal de comprovar que les funcionalitats dels components continuen funcionant i que no hi ha cap element inestable. Amb les reunions amb el director del projecte també s'aniran validant les funcions del dispositiu i comprovar que s'han complert els objectius.

3 Planificació temporal

3.1 Durada estimada del projecte

Aquest projecte es preveu realitzar-se aproximadament en 4 mesos, de mitjans de setembre de 2016 fins a principi de gener de 2017.

3.2 Descripció de les tasques

3.2.1 Fita inicial: GEP

La primera tasca consisteix en cursar l'assignatura Gestió de Projectes (GEP) de tres crèdits ECTS. En aquesta tasca es defineixen els aspectes més importants del projecte i es realitza documentació essencial per un projecte la qual serà inclosa en la memòria final del projecte.

3.2.2 Compra de components

Aquesta tasca consisteix en la compra de components i eines necessàries per la construcció del projecte. Es cerquen els components en diferents botigues online i es fa una comparativa segons el preu i el temps d'enviament, que interessa que sigui relativament petit per tal de no endarrerir el projecte, es trien les botigues que millor satisfacin aquestes condicions.

3.2.3 Configuració de l'entorn de treball

Els programes que es faran servir per a programa són el software d'Arduino[7] (versió 1.6.12) i el software Code Composer Studio[8] (versió 6) proporcionat per Texas Instruments. La instal·lació i configuració del CCS pot resulta complicada, es farà amb l'ajuda d'una guia de l'assignatura vista en el grau.

3.2.4 Desenvolupament del dispositiu

El desenvolupament del projecte es divideix en 3 tasques principals. La programació de l'Arduino, la programació del DSP i la comunicació entre l'Arduino i el DSP.

3.2.4.1 Programació de l'Arduino

Descripció:

La programació de l'Arduino consisteix en la configuració de la lectura de polsadors i potenciòmetres, la configuració de la pantalla LCD i la configuració del guardat de patrons.

Alternativa, pla d'acció i possibles desviacions:

La lectura correcta dels polsadors i dels potenciòmetres és indispensable que funcioni, no és una tasca complicada i no hauria d'haver complicacions. El que pot donar problemes són els rebots dels polsadors i la solució serà idear un codi que els tingui en compte. Respecte la pantalla LCD, és important que funcioni ja que és l'única manera de veure les notes seleccionades i serà de gran utilitat per l'usuari, tot hi així no és un part indispensable i si sorgeixin problemes es buscaria un altre model de pantalla o s'obviaria del projecte. Per guardar els patrons s'utilitza la memòria interna no volàtil EPROM de l'Arduino.

Recursos:

Hardware: Arduino Mega 2560, dos matrius de polsadors, varis potenciòmetres, una pantalla LCD, varis cables

Software: Windows 7, Arduino 1.6.12

3.2.4.2 Programació del DSP

Descripció:

La programació del DSP consisteix en la generació i seqüenciació de les notes i la configuració dels efectes de so.

Alternativa, pla d'acció i possibles desviacions:

Per la generació de so no hi ha d'haver problemes ja que s'ha vist anteriorment en l'assignatura del grau i es tenen exemples de generació d'ones. Per la seqüenciació s'ha d'idear un mecanisme per a que les notes sonin correctament i en ordre. Un problema que pot sorgir en la qualitat de l'àudio són els *clicks*, s'investigarà sobre síntesi musical per saber com evitar-ho.

Recursos:

Hardware: C5515 eZDSP USB Stick Development Tool

Software: Windows 7, Arduino 1.6.12 , Code Composer Studio

3.2.4.3 Comunicació Arduino amb el DSP**Descripció:**

Aquesta tasca consisteix en comunicar l'Arduino amb el DSP mitjançant la interconnexió I2C.

Alternativa, pla d'acció i possibles desviacions:

Si el bus I2C no funcionés, es podria mirar de generar l'àudio amb el propi Arduino i que l'enviés per un jack al DSP. Tot i que aleshores el Arduino podria no tenir suficient còmput en temps real per realitzar les tasques de lectura i interacció.

Recursos:

Hardware: Arduino Mega 2560, C5515 eZDSP USB Stick Development Tool, cables de *wirewrap*.

Software: Windows 7, Code Composer Studio

3.2.5 Construcció de la caixa

La idea de la caixa és que només es vegin els components per interactuar, és a dir, els polsadors, els potenciòmetres i la pantalla LCD. També requerirà d'un forat per al cable USB que farà d'alimentació del sistema.

Tot i que la idea original de la caixa era fer-la de fusta i a mida, s'ha optat per comprar una caixa d'alumini ja feta i adaptar-la al sistema i fer els forats necessaris.

L'aspecte i construcció no són dels factors més importants del projecte i per tant tindran una prioritat menor respecte les altres tasques.

3.2.6 Proves i verificació

Aquesta tasca consisteix en la realització de proves del producte, per tal de verificar el correcte funcionament del sistema. Per cada tasca desenvolupada es realitzarà un testejat del sistema per controlar que tot continua funcionant. Un cop finalitzat tot el sistema es realitzarà un testejat final i exhaustiu de tot el conjunt. Això s'ha de fer abans de posar el sistema a la caixa ja que si s'hagués de canviar alguna connexió es faria més complicat.

3.2.7 Fita final

Per finalitzar el projecte, s'ha de redactar la memòria final on s'expliqui tot el procés del projecte. La memòria es va realitzant a mida que s'avança el projecte per tal de tenir-ho tot ben documentat i no s'oblidin parts.

Per tant aquesta tasca consistirà en recopilar tota aquesta documentació, revisar-la i unir-la en un únic document.

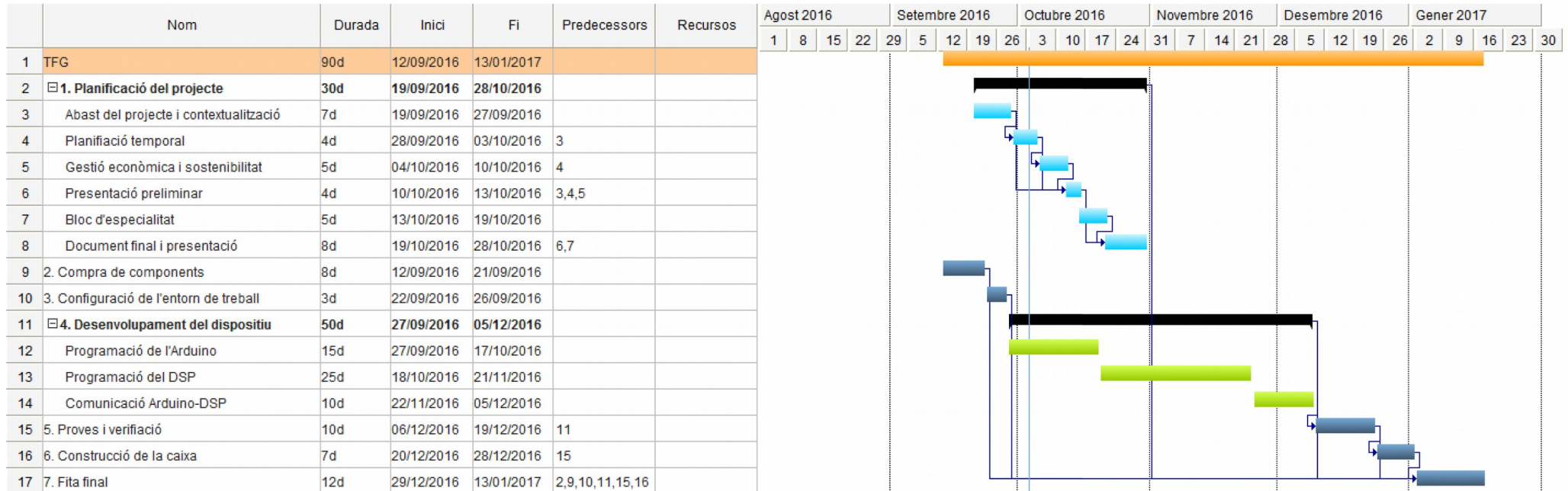
3.3 Recursos humans

Aquest projecte serà desenvolupat per una sola persona, la qual és encarregada de realitzar tots les rols que hi ha en el projecte:

- Director del projecte (DP): realitza les investigacions sobre el mercat actual, tria els dispositius i programari a utilitzar i redacta els informes finals del projecte.
- Programador Arduino (PA): és l'encarregat de desenvolupar les funcions de l'Arduino i la interconnexió de components necessaris.
- Programador DSP (PD): és l'encarregat de desenvolupar el software pel DSP.
- Programador comunicació Arduino-DSP (PC): és l'encarregat de desenvolupar el protocol de comunicació entre l'Arduino i el DSP i de les interconnexions que facin falta.
- Tester (TE): és l'encarregat de testejar el sistema i verificar que tot funciona com s'espera.
- Dissenyador (DI): és l'encarregat de dissenyar la caixa i distribuir els diferents components d'una forma ergonòmica.

3.4 Diagrama de Gantt

La programació establerta és la següent, com es pot observar no hi ha dependències en el desenvolupament del dispositiu, per tant és possible reordenar aquestes tasques.



Il·lustració 3.1: Diagrama de Gantt

4 Gestió econòmica

4.1 Identificació i estimació dels costos

En aquest apartat es realitza una estimació dels costos econòmics que té el projecte, classificant-los en tres tipus: directes (recursos humans, hardware i software) i indirectes.

4.1.1 Costos directes

4.1.1.1 Costos en recursos humans

Com s'ha esmentat anteriorment, aquest projecte serà desenvolupat per una sola persona la qual s'encarrega de tots els rols del projecte esmentats en l'apartat 3.3 Recursos Humans. La següent taula mostra el pressupost estimat en recursos humans separat per rols.

Rol	Hores	€/h	Cost
DP	120	50,00 €	6.000,00 €
PA	64	25,00 €	1.600,00 €
PD	110	35,00 €	3.850,00 €
PC	45	30,00 €	1.350,00 €
TE	40	15,00 €	600,00 €
DI	28	20,00 €	560,00 €
TOTAL			13.960,00 €

Taula 4.1: Costos dels recursos humans per rols

A continuació es mostra el pressupost per nivell de tasques del diagrama de Gantt.

Nom de la tasca	Hores estimades	Recursos	Cost
Planificació del projecte	88		4.400,00 €
Abast del projecte i contextualització	28	DP	1.400,00 €
Planificació temporal	10	DP	500,00 €
Gestió econòmica i sostenibilitat	10	DP	500,00 €
Presentació preliminar	10	DP	500,00 €
Bloc de l'especialitat	10	DP	500,00 €
Document final i presentació	20	DP	1.000,00 €
Compra de components	2	DP	100,00 €
Configuració de l'entorn de treball	20	PA/PD/PC	600,00 €
Desenvolupament del dispositiu	200		6.200,00 €
Programació de l'Arduino	60	PA	1.500,00 €
Programació del DSP	100	PD	3.500,00 €
Comunicació Arduino-DSP	40	PC	1.200,00 €
Proves i verificació	40	TE	600,00 €
Construcció de la caixa	28	DI	560,00 €
Fita final	30	DP	1.500,00 €
Total	408		13.960,00 €

Taula 4.2: Costos en recursos humans per tasques

4.1.1.2 Costos Hardware

En aquest apartat es calcula el cost del hardware necessari per a la programació del sistema i per al muntatge d'aquest.

Es calcula l'amortització tenint en compte que la durada del projecte és de 5 mesos amb la següent expressió:

$$Amortització = \frac{Preu}{Vida\ útil} \cdot \frac{5\ mesos}{12\ mesos/any}$$

Producte	Preu	Vida útil (anys)	Amortització
Ordinador sobretaula	1.350,00 €	5	112,50 €
Amplificador (altaveus)	250,00 €	10	10,42 €
Eina wire-wrap	10,00 €	10	0,42 €
Soldador	20,00 €	5	1,67 €
Protoboard	6,00 €	10	0,25 €
Cable wire-wrap	5,00 €	15	0,14 €
TOTAL			125,39 €

Taula 4.3: Costos hardware indirecte

I aquest és el pressupost dels components hardware necessaris pel sistema.

Producte	Preu
Arduino Mega 2560	7,08 €
C5515 eZDSP	70,80 €
Potenciòmetres (10 unitats)	4,56 €
Knobs (10 unitats)	2,41 €
Teclat polsadors AD (2 unitats)	8,68 €
Adaptador SAMTEC	6,86 €
Pantalla LCD	2,63 €
Hub USB	3,50 €
Caixa	40,00 €
TOTAL	146,52 €

Taula 4.4: Costos hardware components

4.1.1.3 Costos Software

L'amortització es calcula tenint en compte que la durada del projecte és de 5 mesos.

Software	Preu	Vida útil (anys)	Amortització
Code Compose Studio	0,00 €		0,00 €
Arduino Software	0,00 €		0,00 €
Windows 7 Profesional	16,00 €	3	2,22 €
Adobe Photoshop CS5	60,45 €	1	25,19 €
TOTAL			27,41 €

Taula 4.5: Costos software

4.1.2 Costos indirectes

En aquests costos es contempla el lloguer d'una oficina i l'electricitat consumida durant el transcurs del projecte. Es suposa que el consum de l'ordinador és de 400W i el preu de l'electricitat de 0.124 €/kWh.

Producte	Unitat	Preu	Preu total
Lloguer d'una oficina	5 mesos	450 €/mes	2.250,00 €
Servei Internet	5 mesos	50 €/mes	250,00 €
Electricitat	2 kWh per dia durant 5 mesos	0.124 €/kWh	37,20 €
TOTAL			2.537,20 €

Taula 4.6: Costos indirectes

4.1.3 Imprevistos

Per calcular els costos d'imprevistos es suposa que hi ha un risc del:

- 10% a que s'endarrereixi el projecte 15 dies

$\text{Cost retard} = 0,1 * (\text{cost recursos humans} + \text{costs directes} + \text{costs indirectes}) *$

$15 \text{ dies} / 5 \text{ mesos} = 0,1 * 16796,52 \text{ €} * 15\text{d} / 150\text{d} = 167,97 \text{ €}$

- 10% que s'espatlli alguna peça de hardware

$\text{Cost averia} = 0,1 * (\text{cost hardware indirecte} + \text{cost hardware components}) =$

$0,1 * 271,91 = 27,19 \text{ €}$

Imprevist	Cost
Retard 15 dies	167,97 €
Averia	27,19 €
TOTAL	195,16 €

Taula 4.7: Costos per imprevistos

4.1.4 Contingència i cost total

Es suma un 10% al total del pressupost calculat com a mesura de contingència. El cost total queda de la següent forma:

Tipus de cost	Cost
Cost recursos humans	13.960,00 €
Costos hardware i software	299,32 €
Costos indirectes	2.537,20 €
Costos per imprevistos	195,16 €
Cost parcial	16.991,68 €
Contingència (10% del cost parcial)	1.699,17 €
TOTAL	18.690,85 €

Taula 4.8: Cost total

4.2 Control de gestió

A mida que avanci el projecte es podran calcular els costos reals de cada tipus de cost i les hores reals de treball de recursos humans. Per tant podrem determinar les desviacions del projecte amb els següents indicadors:

- Desviació temporal = Data finalització estimada tasca – Data finalització real tasca
- Desviació cost d'una tasca = Cost estimat tasca – Cost real tasca
- Desviació tipus de cost = Cost tipus estimat – Cost tipus real
- Desviació econòmica total = Cost total previst – Cost total real

4.3 Cost Final

Un cop finalitzat el projecte s'ha realitzat una comparació del pressupost estimat i del pressupost real. És força complicat calcular exactament les hores invertides en el projecte, però es considera que s'han realitzat les hores estimades. Com no hi ha hagut retards en la programació ni problemes s'han eliminat els costos per imprevistos i el cost de contingència. La diferència entre el cost estimat i el cost final es força semblant, per tant s'ha fet una bona previsió.

Tipus de cost	Cost
Cost recursos humans	13.960,00 €
Costos hardware i software	299,32 €
Costos indirectes	2.537,20 €
Cost total	16.796,52 €

Taula 4.9: Cost final del projecte

4.4 Viabilitat comercial del producte

Tenint en compte que cada producte tindria un cost material aproximat de 145€, es pot fer un càlcul de la quantitat de dispositius que s'haurien de fabricar per obtenir benefici. A un preu de venda de 242€ (200€ sense IVA) s'haurien de vendre aproximadament 300 unitats, és a dir, que a partir d'aquí s'obtindria benefici. Però per a la seva fabricació en massa s'hauria de realitzar un disseny més exhaustiu del producte.

5 Sostenibilitat i compromís social

En aquest apartat es realitzarà un anàlisi de l'impacte del projecte abans, durant i després de la realització d'aquest. S'analitzen tres dimensions, l'ambiental, l'econòmica i la social, i a partir d'aquestes s'elaborarà la matriu de sostenibilitat. Aquesta matriu està basada en el concepte de l'economia del bé comú[11], ideat per Christian Felber.

5.1 Dimensió ambiental

El desenvolupament d'aquest projecte tindrà un impacte ambiental baix respecte el consum d'electricitat. Tot i que el DSP i l'Arduino consumeixen electricitat, aquesta és molt baixa respecte la que consumeix l'ordinador sobretaula, que és el component que més consumeix.

Per altra banda els components de les plaques electròniques tenen un petit impacte en la seva fabricació, ja que es produeixen en massa, i un gran impacte quan aquests components es converteixen en residus, és a dir quan ja no realitzen la seva funció, per tant s'han de portar a la planta de tractament de residus electrònics adient o és pot optar per dessoldar-los i guardar-los per a la seva reutilització.

5.2 Dimensió econòmica

En apartats anteriors s'ha realitzat un pressupost detallat on també s'han inclòs sobre costos de possibles imprevistos. La conclusió és que el projecte és econòmicament viable.

El costos dels components hardware és poc notable, però tot així poden ser reduïts si es realitza la compra de components amb més antelació a botigues online on els components són molt més barats però el temps d'enviament més gran. O la compra de clons, com en el cas de l'Arduino Mega 2560. Els costos software no suposen impacte econòmic ja que són programes gratuïts, excepte el sistema operatiu.

Els costos indirectes són més notables i no es poden reduir atès que són indispensables pel desenvolupament del projecte.

Els costos de recursos humans són els més notables, això és degut a la durada del projecte i les diverses tasques específiques segons el diagrama de Gantt establert a la

documentació. Hi ha la possibilitat de realitzar el projecte en menys temps si es contractessin més programadors ja que es poden realitzar les tasques en paral·lel però això generaria un increment del cost.

5.3 Dimensió social

És un projecte que no té un impacte social. No afecta a la població en general, només pot afectar a aquells que es dediquin a la producció musical.

5.4 Matriu de sostenibilitat

	Projecte en producció	Vida útil	Riscs
Ambiental	Consum del disseny	Petjada ecològica	Riscs ambientals
	8 (0:10)	17 (0:20)	0 (-20:0)
Econòmic	Factura	Viabilitat econòmica	Riscs econòmics
	8 (0:10)	17 (0:20)	0 (-20:0)
Social	Impacte personal	Impacte social	Riscs socials
	8 (0:10)	12 (0:20)	0 (-20:0)
Rang de sostenibilitat	24 (0:30)	46 (0:60)	0 (-60:0)
	70 (-60:90)		

Taula 5.1: Matriu de sostenibilitat

6 Fonaments teòrics

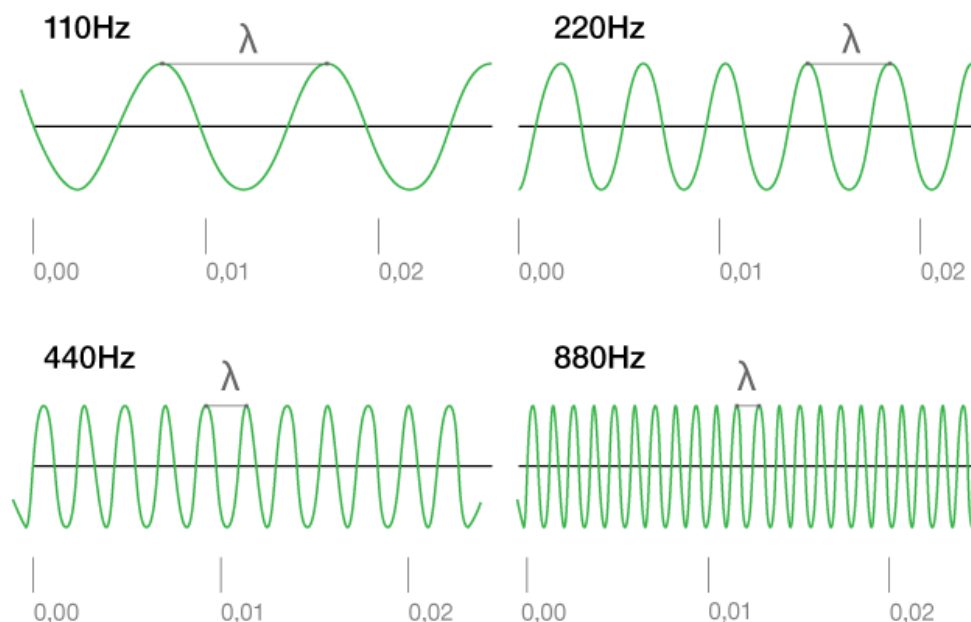
6.1 So

En la física, el so és una vibració que es propaga com una ona mecànica a través d'un medi com pot ser l'aire o l'aigua. En la psicologia, el so és la recepció d'aquestes ones i la percepció que en fa el cervell. Els ésser humans poden sentir les ones soneres amb freqüències compreses entre els 16 Hz i 20 kHz.[12]

6.1.1 Propietats bàsiques

6.1.1.1 Freqüència

La freqüència d'una ona sonora, també anomenada altura en el món de la música, determina si un so serà greu (baixes freqüències, vibració lenta) o serà agut (altes freqüències, vibració alta). Com més petita la longitud d'ona (λ), més alta serà la freqüència, de la mateixa manera com és gran sigui la longitud d'ona, més baixa serà la freqüència[13].



Il·lustració 6.1: Ones de diferent freqüències

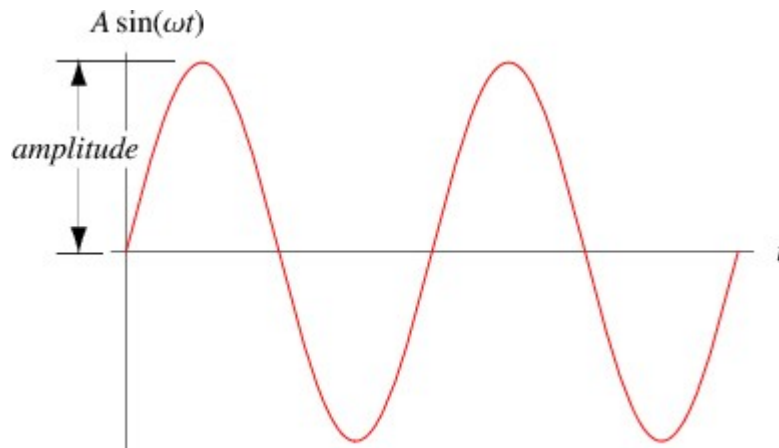
Les notes musicals estan separades per semitons, és a dir, cada nota té un freqüència diferent. Si es pren la freqüència 440 Hz com a referència pel La3 (la nota La de l'escala tercera) les diferents freqüències queden de la següent manera[14]:

Nota	Freqüència (Hz)	Nota	Freqüència (Hz)
C0	16.35	F #4/Gb4	369.99
C#0/Db0	17.32	G4	392.00
D0	18.35	G#4/Ab4	415.30
D#0/Eb0	19.45	A4	440.00
E0	20.60	A#4/Bb4	466.16
F0	21.83	B4	493.88
F#0/Gb0	23.12	C5	523.25
G0	24.50	C#5/Db5	554.37
G#0/Ab0	25.96	D5	587.33
A0	27.50	D#5/Eb5	622.25
A#0/Bb0	29.14	E5	659.25
B0	30.87	F5	698.46
C1	32.70	F#5/Gb5	739.99
C#1/Db1	34.65	G5	783.99
D1	36.71	G#5/Ab5	830.61
D#1/Eb1	38.89	A5	880.00
E1	41.20	A#5/Bb5	932.33
F1	43.65	B5	987.77
F#1/Gb1	46.25	C6	1046.50
G1	49.00	C#6/Db6	1108.73
G#1/Ab1	51.91	D6	1174.66
A1	55.00	D#6/Eb6	1244.51
A#1/Bb1	58.27	E6	1318.51
B1	61.74	F6	1396.91
C2	65.41	F#6/Gb6	1479.98
C#2/Db2	69.30	G6	1567.98
D2	73.42	G#6/Ab6	1661.22
D#2/Eb2	77.78	A6	1760.00
E2	82.41	A#6/Bb6	1864.66
F2	87.31	B6	1975.53
F#2/Gb2	92.50	C7	2093.00
G2	98.00	C#7/Db7	2217.46
G#2/Ab2	103.83	D7	2349.32
A2	110.00	D#7/Eb7	2489.02
A#2/Bb2	116.54	E7	2637.02
B2	123.47	F7	2793.83
C3	130.81	F#7/Gb7	2959.96
C#3/Db3	138.59	G7	3135.96
D3	146.83	G#7/Ab7	3322.44
D#3/Eb3	155.56	A7	3520.00
E3	164.81	A#7/Bb7	3729.31
F3	174.61	B7	3951.07
F#3/Gb3	185.00	C8	4186.01
G3	196.00	C#8/Db8	4434.92
G#3/Ab3	207.65	D8	4698.63
A3	220.00	D#8/Eb8	4978.03
A#3/Bb3	233.08	E8	5274.04
B3	246.94	F8	5587.65
C4	261.63	F#8/Gb8	5919.91
C#4/Db4	277.18	G8	6271.93
D4	293.66	G#8/Ab8	6644.88
D#4/Eb4	311.13	A8	7040.00
E4	329.63	A#8/Bb8	7458.62
F4	349.23	B8	7902.13

Taula 6.1: Freqüències de les notes musicals

6.1.1.2 Amplitud

L'amplitud determina la quantitat d'energia que transporta una ona. Com més o menys energia, més o menys potència tindrà l'ona i per tant el so el percebrem més fort o més fluix. Com més alta l'amplitud, més intens és el so[15].



Il·lustració 6.2: Amplitud d'una ona sinusoïdal

El decibel (dB) és la unitat de mesura de la potència o intensitat del so, tot i que és una relació entre un so determinat i el de referència de 0 dB. L'escala de decibels és logarítmica, en què una duplicació de la potència provoca un augment aproximat de 3 dB[16].

6.2 Àudio digital

L'àudio digital és la tecnologia usada per gravar, emmagatzemar, generar, manipular i reproduir so utilitzant senyals d'àudio que han estat codificats digitalment[17].

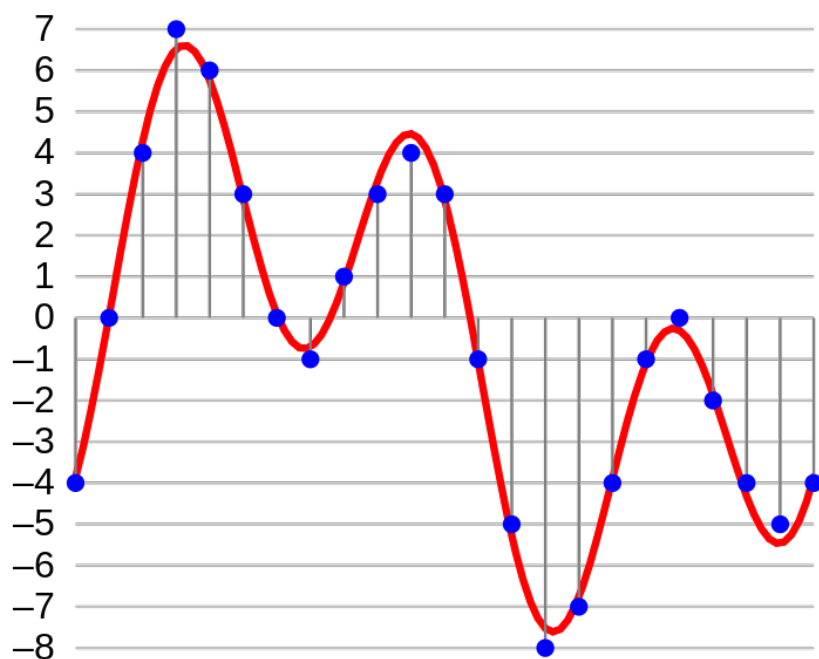
6.2.1 Mostreig

El mostreig digital consisteix en prendre mostres d'una senyal analògica, com pot ser el so, amb un freqüència constant (freqüència de mostreig) i quantificar cada mostra.

6.2.1.1 Freqüència de mostreig

El teorema de Nyquist formula que la freqüència de mostreig mínima que es requereix per realitzar una gravació digital de qualitat, ha de ser superior al doble de la freqüència d'àudio del senyal analògic que s'intenta digitalitzar[18].

Com que l'oïda humana percep els sons d'entre 20 i 20.000 Hz, en el món de l'àudio, la freqüència que més es fa servir és la de 44.100 Hz (CD) ja que permet guardar o reproduir l'àudio digital amb la fidelitat suficient per a no perdre informació rellevant.



Il·lustració 6.3: Mostreig d'una ona amb profunditat de 4 bits

6.2.1.2 Resolució

La resolució, també anomenat profunditat de bits, és el nombre de bits utilitzats per emmagatzemar cada mostra del senyal analògic. Les resolucions que més s'utilitzen en el àudio digital són de 8 bits, 16 bits i 32 bits. Mentre que una resolució de 8 bits proporciona 256 (2^8) nivells d'amplitud, una de 16 bits arriba a 65536 (2^{16}). La profunditat de bits dels CD's són de 16 bits (estèreo).

6.2.2 Conversió analògic – digital

La conversió analògic-digital consisteix bàsicament a realitzar de forma periòdica mesures de l'amplitud (tensió) d'un senyal, arrodonir els valors a un conjunt finit de nivells preestablerts i registrar-los com a nombres enters (quantificar-los). Les fases de la digitalització són el mostreig, la retenció (*hold*), la quantificació i la codificació[19].

Per exemple, l'Arduino Mega 2560 disposa de 15 canals d'entrada analògica que poden ser convertits a valors digitals. Realment no disposa de 15 ADC diferents, sinó que en té només un amb una entrada multiplexada de manera que es selecciona el canal que es vol llegir amb el multiplexador.

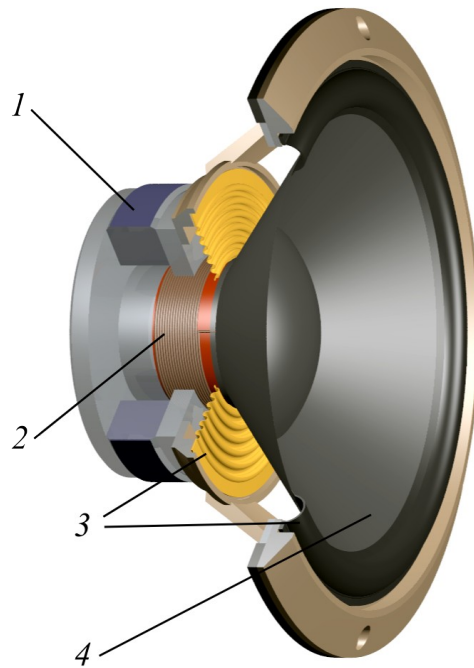


Il·lustració 6.4: Ports analògics de l'Arduino Mega 2560

6.2.3 Conversió digital – analògic

Consisteix en el procés invers de l'apartat anterior. En processa informació binària (valors concrets) i es converteix a un valor de tensió analògic (valors continus) que son processats per un circuit electrònic que interpreta el senyal [20], en el cas de l'àudio (altaveus) transforma el senyal de tensió en so.

El senyal elèctric arriba a l'imant que per efecte electromagnètic fa moure la bobina del seu interior, el moviment vibratori de la bobina es transmet al diafragma que genera les ones mecàniques audibles.



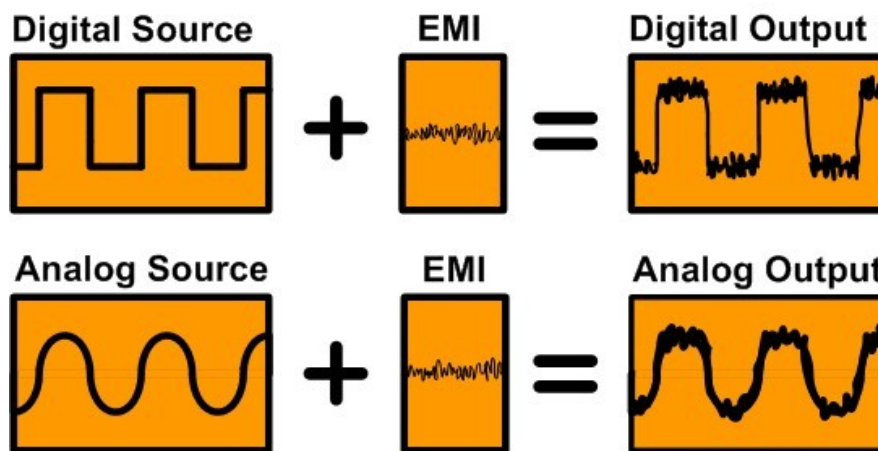
Il·lustració 6.5: Esquema d'un altaveu. 1: imant, 2: bobina, 3: suspensió, 4: diafragma

6.2.4 Soroll

Les interferències electromagnètiques (EMI) són pertorbacions generades per una font externa o per algun component intern del mateix sistema que afecten a l'equip electrònic. Aquestes pertorbacions poden crear problemes de funcionament del sistema incloent l'aturament o la pèrdua de dades.

Les interferències electromagnètiques poden ser ocasionades per un motor, un interruptor d'engegada, llums fluorescents, inclús per la pròpia persona... Les EMI generades es mesuren en volts/metre (V/m) i la relació amb el corrent elèctric del circuit és lineal i directe[21].

Existeixen varies mesures per eliminar o reduir les interferències electromagnètiques com poden ser la utilització de condensadors de desacoblament, usar un clock diferencial (elimina el soroll del clock original), la gàbia de Faraday, etc... o simplement reduir la llargada de les connexions, reduint la impedància dels borns de connexió i fent un bon disseny de la senyal terra del circuit[21].



Il·lustració 6.6: Exemple de la pertorbació que pot generar un EMI en l'àudio

7 Elecció i descripció dels components

En aquesta secció es justifica l'elecció dels components que han estat escollits per a fabricar el seqüenciador de notes musicals. També se'n farà una breu descripció de les especificacions de cadascun i se'n contemplaran les limitacions.

7.1 Arduino Mega 2560

L'Arduino Mega 2560[5] està basat en el microcontrolador ATmega2560, un microcontrolador de baix consum de 8-bit i d'arquitectura RISC de l'empresa Atmel. La placa d'Arduino disposa de 54 pins digitals d'entrada o de sortida dels quals 15 poden ser configurats com a PWM, també disposa de 16 entrades analògiques (10 bits de resolució), 4 ports UART, un oscil·lador de 16 MHz, connexió USB, un connector ICSP i un botó de reinici.

El ATmega2560 ve preprogramat amb un bootloader que permet pujar nou codi a la placa sense la necessitat de fer servir un programador hardware extern, gràcies a la utilització del protocol de comunicació STK500 de Atmel. Tot i que si és necessari es pot fer servir el connector ICSP.

El Mega 2560 té un fusible que protegeix els ports USB de l'ordinador de sobrecàrregues i curtcircuits. Tot i que els ordinadors ja tenen la seva pròpia protecció interna, el fusible ens proporciona una capa addicional de protecció. Si s'apliquen més de 500 mA al port USB, el fusible tallarà la corrent fins que desaparegui la sobrecarrega o curtcircuit que l'ha ocasionat.

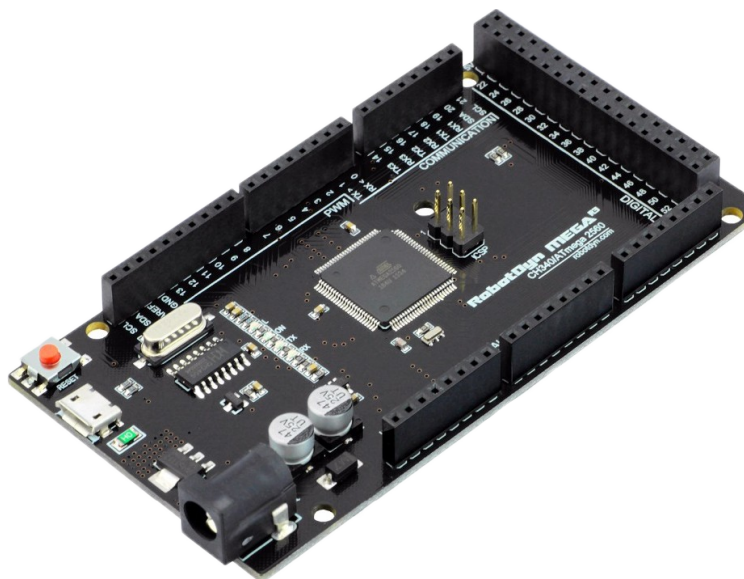
L'Arduino es pot alimentar mitjançant la connexió USB o mitjançant una font d'alimentació externa. L'alimentació externa pot ser d'un convertidor de corrent altern a continua. L'Arduino pot operar de 6 a 20V. Tot i així si es subministren menys de 7V, el pin 5V pot no proporcionar tot el seu valor i la placa pot estar inestable; en canvi, si es subministren més de 12V el regulador es pot escalfar massa i pot danyar els components electrònics de la placa. Per tant es recomana subministrar de 7 a 12V.

Els 54 pins digitals poden ser configurats tant d'entrada com de sortida. Cada pin pot rebre i subministrar 20 mA i disposen d'una resistència pull-up de 20-50 k Ω que està desconnectada per defecte. Es recomana que no es passi de 40 mA per evitar danys a la placa. Disposada de múltiples pins serial (RX, TX), pins per a interrupcions externes, pins de

PWM de 8 bits, pins SPI (MISO, MOSI, SCK, SS) i pins TWI (SDA, SCL). Tots aquests pins disposen de llibreries open-source proporcionades per Arduino (SoftwareSerial.h, SPI.h, Wire.h, etc..) que faciliten la seva programació, també disposen de programes d'exemples que faciliten la seva entesa.

Quan es va començar el projecte es va utilitzar un Arduino Uno que té les mateixes característiques que el Mega 2560 però disposa de menys pins analògics i per tant no tenia suficients pins. Per això es va optar per fer servir un Arduino Mega 2560, ja que disposa de multitud de pins i l'únic inconvenient que té seria que ocupa més espai físic.

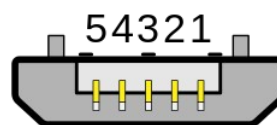
Com s'ha comentat anteriorment, s'utilitzarà un clon de l'Arduino Mega 2560. Té el mateix microcontrolador (ATmega2560) i té les mateixes característiques excepte que no té una connexió per USB sinó per micro-USB de tipus B.



Il·lustració 7.1: Arduino Mega 2560 R3 CH340G
(clon)



Il·lustració 7.2: Connector micro
USB Tipus B mascle



Micro-B

Il·lustració 7.3: Connexió
micro USB tipus B

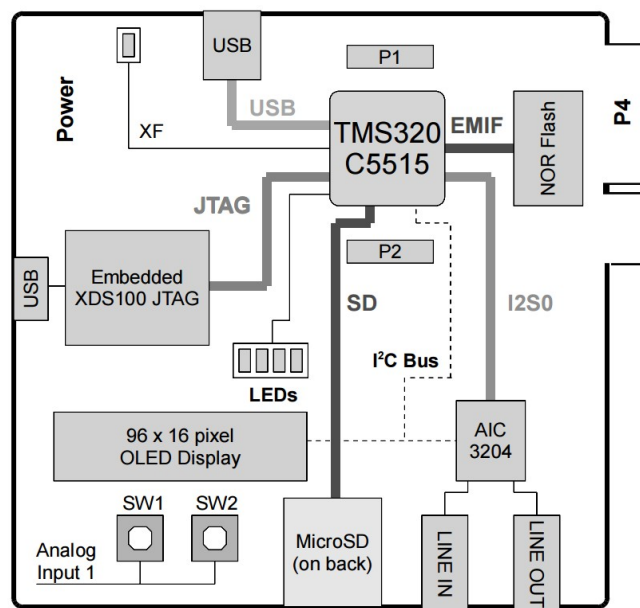
7.2 TMS320C5515 eZdsp USB Stick

El TMS320C5515 eZdsp USB Stick[22][23] (a partir d'ara 'C5515 USB Stick' per abreviar) és un producte de Texas Instrument dissenyat per Spectrum Digital. El C5515 USB Stick és una placa amb un dels DSP de més baix cost i de més baix consum de la indústria. És un dispositiu molt complet pel preu que presenta ja que es pot fer servir en molts camps del processament de senyals, ja pot ser la gravació d'àudio i el seu tractament, instruments musicals, aplicacions mèdiques mòbils o altres aplicacions en el camp de la indústria en general o la seguretat. Consta del processador TMS320C5515 i els seus perifèrics, i el còdec TLV320AIC3204.

7.2.1 Processador TMS320C5515

El C5515 USB Stick consta del microprocessador de coma fixa TMS320C5515. Es pot configurar el PLL per a fer-lo funcionar a la freqüència desitjada; les més usuals són 60, 75, 100 o 120 MHz. Disposa de 320 Kbytes de memòria RAM i 128 Kbytes de memòria ROM.

A més, el processador disposa de diferents perifèrics: una ranura microSD, un port miniUSB, una memòria NOR flash de 32 Mbytes, una pantalla OLED I2C de 96 x 16 píxels, 4 LEDs programables i 1 d'encesa, 2 polsadors, un connector USB amb l'emulador XDS100 JTAG que permet debugar des del programari del PC, dos connectors d'interfície Bluetooth (P1, P2), un slot d'expansió propi amb diferents senyals (P4), i el més rellevant, el còdec TLV320AIC3204.



Il·lustració 7.4: Diagrama de blocs del C5515 eZdsp USB Stick

7.2.2 TLV320AIC3204

El TLV320AIC3204[24] és un còdec de baix consum que disposa d'un convertidor digital a analògic (DAC) estèreo i d'un convertidor analògic a digital (ADC) estèreo. Es comunica amb el C5515 mitjançant I2C i per transmetre dades de l'àudio fa servir una tecnologia de comunicació semblant, I2S. L'AIC3204 pot reproduir àudio a una freqüència de fins a 48000 Hz, és a dir 24000 mostres al segon (és estèreo, per tant ha d'haver-hi una mostra per cada canal, el dret i l'esquerra), de manera que és suficient per a l'oïda humana que a partir de 20000 Hz pràcticament no percep cap so.

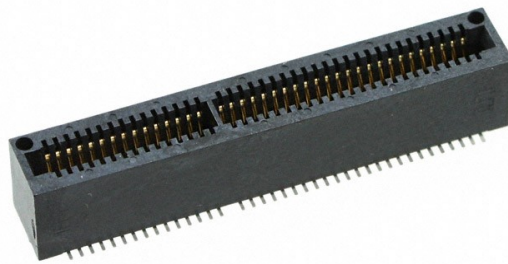
El TLV320AIC3204 té connectats dos minijacks de 3.5 mm, un d'entrada i un de sortida. En aquest projecte només s'utilitza la de sortida ja que només volem reproduir so, no gravar-lo. Per tant caldrà connectar els altaveus al minijack de sortida.



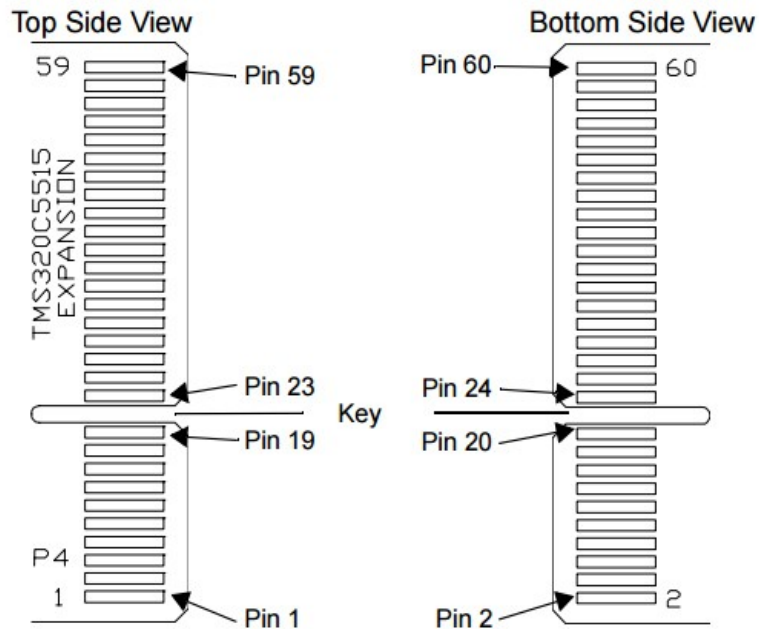
Il·lustració 7.5: Minijack mascle i minijack femella de 3.5 mm

7.2.3 Slot d'expansió

Una altre element rellevant és el slot d'expansió que disposa la placa. Per fer-lo servir es necessita un adaptador, s'ha fet servir el model Samtec MEC1-130-02-S-D-A[25]. En total té 60 pins dels quals hi ha les senyals d'I2C, SDA i SCL, que són les que es fan servir per connectar el C5515 amb l'Arduino. També disposa de varies senyals terra, convé connectar la senyal terra dels dos dispositius per tal de que el nivell de terra sigui el mateix per ambdós.



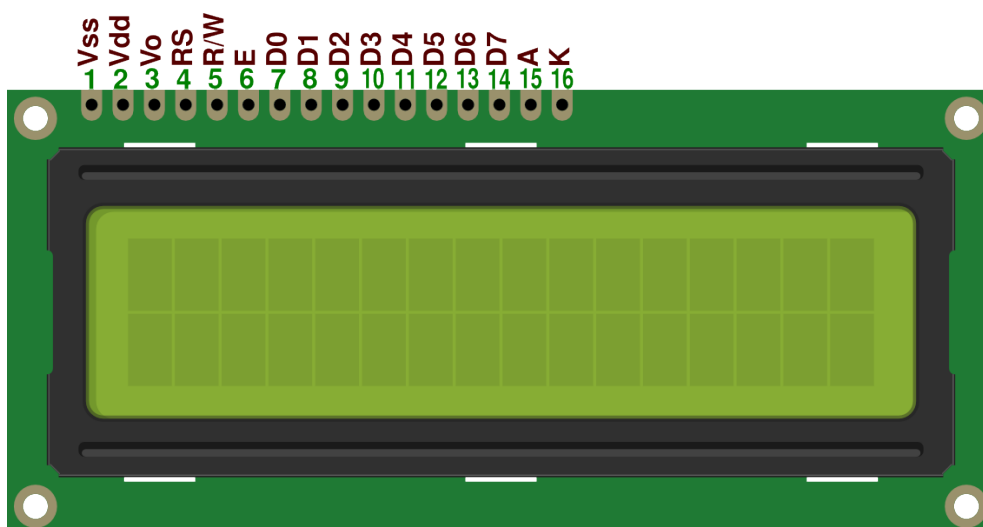
Il·lustració 7.6: Adaptador Samtec MEC1-130-02-S-D-A



Il·lustració 7.7: Slot d'expansió del C5515 USB Stick
(vista per davant i darrere)

7.3 Pantalla LCD 16x2

Per a mostrar la configuració en temps real del seqüenciació s'utilitza una pantalla LCD per a poder veure tota la informació. Aquesta pantalla és controlada per l'Arduino. La pantalla consta de dues files de 16 caràcters de 5x7 píxels, amb un espai entre caràcter i caràcter.



Il·lustració 7.8: Pantalla LCD 16x2 i les seves connexions

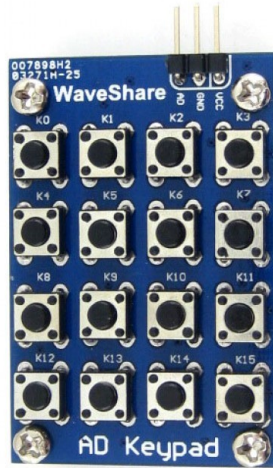
Les connexions RS, E, D4, D5, D6, D7 van a l'Arduino per a poder configurar i controlar la pantalla. El Vo serveix per controlar el contrast de la pantalla, es pot fer servir un potenciòmetre per calibrar-ho però experimentant s'ha trobat que connectant-lo a terra ja té el contrast desitjat. Les connexions A (ànode) i K (càtode) serveixen per encendre o apagar la llum de fons i la llum blanca de les lletres. Per tant la connexió de la pantalla s'ha de realitzar de la següent manera:

Pins Pantalla	
VSS	Terra
VDD	5V
V0	Terra
RS	Pin Arduino (3)
RW	Terra
E	Pin Arduino (4)
D0, D1, D2, D3	No s'usen
D4, D5, D6, D7	Pins Arduino (5,6,7,8)
A	5V amb una resistència de 220 ohms
K	Terra

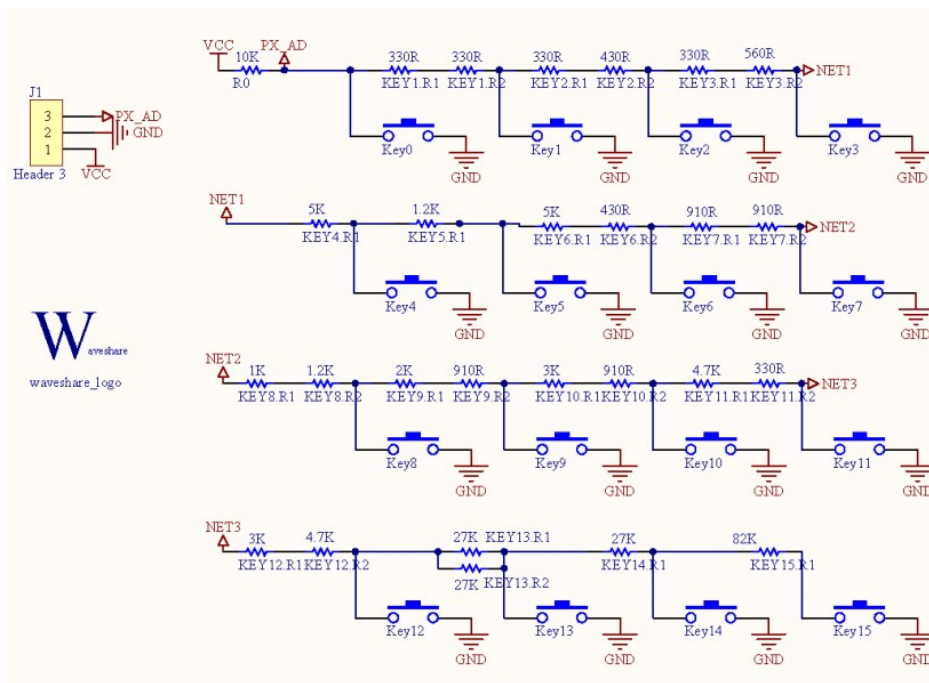
Taula 7.1: Connexions de la pantalla LCD a l'Arduino

7.4 Teclat WaveShare AD

Per a manipular les notes musicals i el seu temps s'utilitzen dos teclats AD fabricats per WaveShare[26]. Disposen de 16 polsadors col·locats en una matriu 4x4. Cada teclat s'alimenta amb 5V. L'avantatge que té aquest teclat és que només ocupa un canal analògic per a la seva lectura. Això és degut a que està format per diferents resistències i segons el polsador que es pitja la tensió que retorna té un valor determinat, per tant només cal comprovar aquest valor i determinar quin polsador s'ha pitjat.



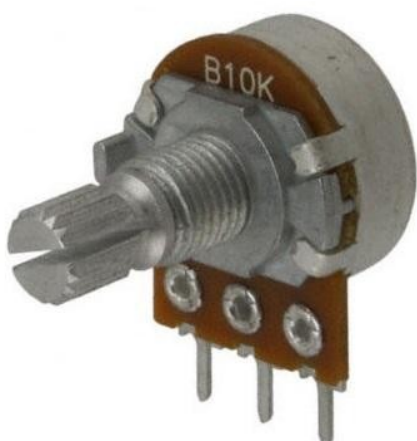
Il·lustració 7.9: AD Keypad



Il·lustració 7.10: Esquema del AD Keypad

7.5 Potenciòmetres

Per a manipular el nivell de volum, el nivell de *pitch* i el nivell de tempo de l'àudio del seqüenciador s'utilitzen tres potenciòmetres. Un potenciòmetre no és res més que una resistència variable. S'alimenta amb 5V i retorna una tensió proporcional segons l'angle que està girat l'eix del potenciòmetre. Els potenciòmetres usats són de 10 k Ω . Per a que siguin més ergonòmics se li han posat knobs de plàstic per a realitzar el gir de l'eix més còmode.



Il·lustració 7.12: Potenciòmetre de 10 k Ω



Il·lustració 7.11: Knobs de plàstic

7.6 HUB USB

Per alimentar el circuit (l'Arduino i el C5515) s'utilitza un HUB USB semblant al mostrat a la il·lustració següent. Amb l'interruptor s'enliga i s'apaga el sistema totalment.



Il·lustració 7.13: HUB USB amb interruptor

7.7 Caixa Retex

Finalment, cal col·locar tots els components dins una caixa robusta i que quedin ben fixats. La primera idea era fer aquest pas amb una caixa de fusta, però amb l'ajuda del director s'ha trobat una millor opció: una caixa de la marca Retex.

Aquestes caixes ja estan pensades per a projectes industrials o projectes que inclouen components electrònics (com pedals de guitarra) i hi ha disponibles un gran ventall de mides. La mida triada de la caixa pel projecte és de 155 x 75 x 175 mm (llargada x altura x ample).

S'hi han realitzat els forats adients per a poder fixar els elements d'interacció. La seva robustesa ve donada per l'alumini anoditzat del que està feta la caixa, per això s'ha folrat amb paper adhesiu transparent per evitar el contacte directe dels components electrònics amb la caixa. També s'ha aprofitat per enganxar-hi papers amb indicacions del que fa cada botó dels teclats i amb un indicador del rang de cada potenciòmetre.

A l'annexe es s'adjunta la construcció de la caixa.

8 Descripció i configuració del software

En aquest apartat s'explica l'estructura del software i les llibreries que s'han utilitzat per a la programació de l'Arduino i del DSP.

8.1 Arduino

S'ha programat l'Arduino per a que llegeixi els polsadors, els potenciòmetre i pinti per pantalla la informació de la seqüenciació de notes actual.

L'estructura del programa de l'Arduino és força senzilla. Consta de dues funcions principals:

```
void setup(){ ... }  
void loop(){ ... }
```

Quan l'Arduino rep alimentació s'executa la funció `setup()` una única vegada. En aquesta funció s'hi realitzen les inicialitzacions i configuracions que calen, com poden ser de la pantalla LCD, la càrrega de dades de la memòria EEPROM o la configuració del bus I2C.

Seguidament s'executa la funció `loop()` que tal i com indica el seu nom s'executa infinitament. Dins d'aquesta funció es troben 4 crides a funcions que tracten els diferents components del sistema: la pantalla, els potenciòmetres i els dos teclats. S'ha comprovat que amb l'estructura Round Robin és suficient per garantir la consistència del sistema. Per a l'enviament de dades al DSP pel bus I2C es crea una crida especial que només s'executa quan se li sol·liciti.

```
void loop(){  
    llegir_Potenciometres();  
    llegir_teclat1();  
    llegir_teclat2();  
    tractar_LCD();  
}
```


8.1.1 Pantalla

Per a la programació de la pantalla s'ha utilitzat la llibreria LiquidCrystal[27]. Consta de varies funcions que permeten inicialitzar i pintar la pantalla còmodement.

Per a la inicialització de la pantalla s'utilitza la funció LiquidCrystal() i begin(). La primera crea una variable del tipus LiquidCrystal i se li ha de passar com a paràmetres els pins que són utilitzats a l'Arduino en el següent ordre: RS, EN, D4, D5, D6, D7. És a dir els pins 2, 3, 4, 5, 6 i 7, en el nostre cas. Per a la segona es passen el nombre de columnes i de files, és a dir, 16 i 2.

```
LiquidCrystal lcd(2, 3, 4, 5, 6, 7);
```

```
...
```

```
lcd.begin(16,2);
```

Per pintar caràcters primer cal usar la funció setCursor(). Se li passa per paràmetre les coordenades on es vol pintar el pròxim caràcter, i seguidament es fa el print() corresponent que pintarà el caràcter on li haguem indicat.

```
lcd.setCursor(13,1);
```

```
lcd.print('0');
```

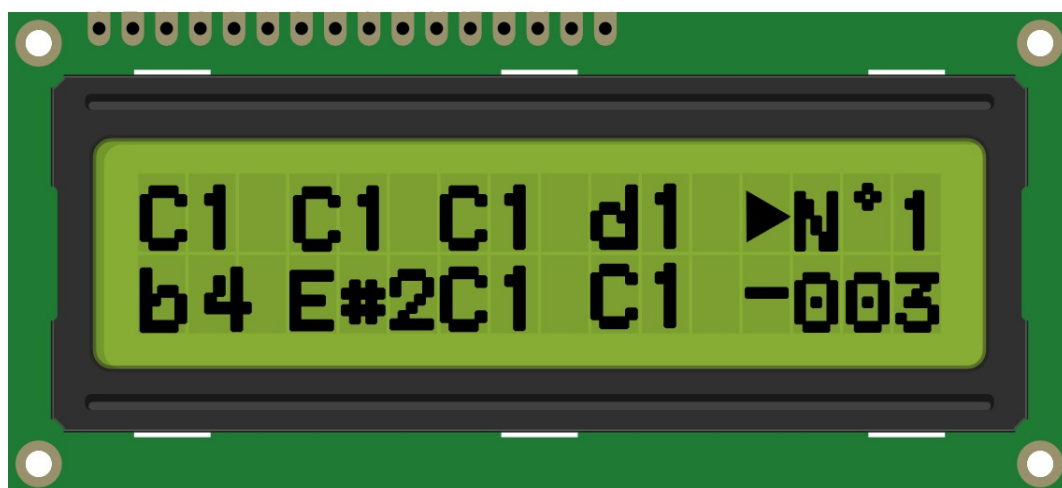
Aquesta pantalla també té una memòria interna que permet guardar fins a 8 caràcters personalitzats. Com el símbol de play i pausa no són reconeguts per la llibreria, s'usen dos caràcters personalitzats que es declaren abans de la funció begin(), amb una funció anomenada createChar() que se li passa com a paràmetre el vector de bytes que formen el caràcter. Exemple:

```
byte play[8] = {B10000, B11000, B11100, B11110, B11100, B11000, B10000};
```

```
lcd.createChar(0, play);
```

Es pensava que en un principi una pantalla 16x2 seria suficient per poder mostrar tota la informació de forma entenedora. Tot i que s'ha aconseguit codificar-la d'alguna forma per a que aparegui tota la informació rellevant i sigui entenedora, la mostra d'informació ha estat bastant limitada per la falta d'espai de caràcters.

Aquest és un exemple del que es pot veure a la pantalla LCD.



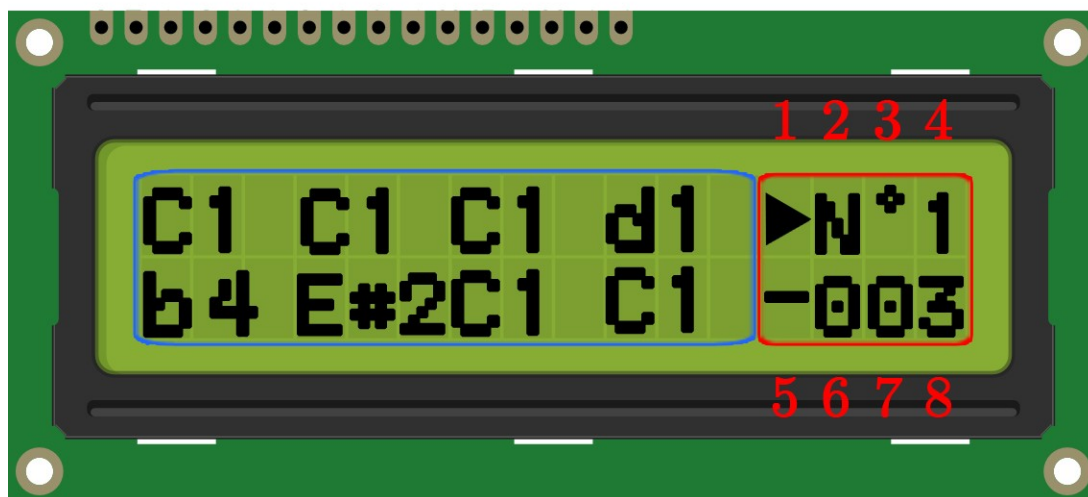
Il·lustració 8.1: Visualització de la pantalla LCD

Consta de dues parts, la part de les notes i la part de l'estat de la configuració. Les notes estan representades per lletres que es com solen representar-ho el anglesos. Les notes do, re, mi, fa, sol, la, si, corresponen a C, D, E, F, G, A, B, respectivament. En la part de les notes (requadre blau de la il·lustració 8.2) es poden veure fins a 8 notes (poden haver-hi entre 1 i 8), i al costat de cada nota tenen un número que correspon a la longitud de la nota. Si la nota és un sostingut, s'indica amb el símbol '#' al costa de la nota. El rang de les notes són l'escala 3 i 4 (les altres són o massa greus o massa agudes) i per representar si una nota està en la 3 o la 4 es posa la nota en minúscula o majúscula, respectivament.

De les columnes 13 a 16 (requadre vermell de la il·lustració 8.2) es fa servir per mostrar els paràmetres de configuració. Cada requadre té una funció específica:

- 1: Mostra si el seqüenciador està en play o en pausa. Per al play mostra el símbol ► i per a la pausa mostra el símbol ||.
- 2: Mostra si la configuració està en mode de modificació de notes (N) o del temps de cada nota (T).
- 3: Quan es modifica un patró de notes, ja sigui canviant una nota o la seva durada, mostra un asterisc (*) en aquest camp per indicar que el patró ha estat modificat. Quan es pitja el botó de guardat (SAVE), l'asterisc es desactiva fins que es torni a modificar el patró actual.
- 4: Indica quin patró està activitat. N'hi ha 7 de disponibles, així que els seus valors són [0,7].

- 5: En aquest camp mostra quin efecte d'àudio està activat. Si no n'hi ha cap mostra un guionet (-), si està activada la reverberació mostra una R i si està activat l'eco mostra una E.
- 6: Aquest camp correspon al pitch general del seqüenciador. Si el potenciòmetre està centrat al mig, el pitch és 0 i mostra un 0. Si el potenciòmetre està girat cap a l'esquerre (negatiu) o la dreta (positiu), mostra un '-' o un '+' respectivament.
- 7: Aquest camp correspon al tempo general del seqüenciador. Si el potenciòmetre està centrat al mig, el tempo és 0 i mostra un 0. Si el potenciòmetre està girat cap a l'esquerre (negatiu) o la dreta (positiu), mostra un '-' o un '+' respectivament.
- 8: Aquest camp correspon al volum general del seqüenciador. El volum es representa amb els valors [0,9], sent el 0 silenci i el 9 el màxim nivell de so.



Il·lustració 8.2: Visualització per camps de la pantalla LCD

8.1.2 Polsadors

Pels polsadors es llegeix el valor que retorna el teclat AD amb la funció `analogRead()` i es determina quin polsador ha estat pitjat. Queda una mica lleig en el codi, ja que són 16 condicions *if* i que en el pitjor dels casos que es polsés l'últim polsador s'haurien de comprovar els 16. Per evitar els *if*'s es va provar de fer servir la funció `map()` d'Arduino que assigna un rang de valors a un altre rang indicat, però ho fa de forma proporcional i els valors que retorna l'AD no són del tot proporcionals per tant amb aquesta funció no es llegia correctament del teclat. Tot i així, els *if*'s no suposen un temps de comput crític que pugui ocasionar inestabilitat o inconsistència al sistema.

Un aspecte a tenir en compte amb els polsadors és l'anomenat efecte rebot[28]. L'efecte rebot és una inestabilitat elèctrica que esdevé quan es pitja i quan es deix de pitjar el polsador. Per solucionar-ho es fa un petit comptador, es compara el valor llegit actual amb el valor anterior i fins aquest fet no és compleix tants cops com el comptador no es determina que ha estat pitjat aquell botó. Es sol contemplar un temps de 20ms fins que el valor es considera estable.

8.1.3 Potenciòmetres

Per llegir els potenciòmetres també s'utilitza la funció `analogRead()`. Com els ports analògics tenen una resolució de 10 bits, retornen un valor entre [0,1023]. Per simplificar per quan s'hagi d'enviar la dada per I2C (s'envien bytes), es divideix entre 4 el valor llegit de forma que un byte sigui suficient per codificar el seu valor.

8.1.4 I2C (Wire.h)

La comunicació de l'Arduino amb el DSP es fa mitjançant la tecnologia de comunicació I2C.

Arduino disposa de la llibreria Wire.h[29] que permet la comunicació I2C amb altres dispositius mitjançant els senyals SDA (Serial Data Line) i SCL (Serial Clock Line).

Per a la comunicació s'escull una adreça de 7 bits, és a dir, del 0 a 127. S'inicia la configuració amb la funció Wire.begin() passant com a paràmetre l'adreça.

Exemple:

```
Wire.begin(0x02);
```

Per a escriure al bus cal cridar la funció Wire.write() que se li pot passar com a paràmetre un vector de bytes i el nombre d'elements a escriure.

Exemple:

```
Wire.write(dataArray,16);
```

S'utilitza l'Arduino com si fos un esclau i el DSP el mestre, per saber l'Arduino quan ha d'enviar les dades es configura amb la funció Wire.onRequest(), que se li passa com a paràmetre la funció que es cridarà quan a l'Arduino se li sol·licitin dades pel bus I2C. És en aquesta funció on es farà l'enviament de dades al DSP.

Exemple:

```
Wire.onRequest(sendDataToDSP);
```

Cada nota està guardada en un byte de 8 bits. Com el rang de les notes és de 24 possibles notes, els 5 bits de menys pes codifiquen la nota. I els 3 bits de més pes es codifiquen pel valor del temps de la nota, que és entre 1 i 8. Si el patró actual és de menys de 8 notes, s'envia un 0x1F en comptes del valor de la nota per indicar que no és vàlida.

En total s'envien 16 bytes en ordre com s'indica a la següent taula:

#Byte	Contingut
Byte 0	Valor del pitch
Byte 1	Valor del tempo
Byte 2	Valor del volum
Byte 3	Indicador de l'efecte de so
Byte 4	Play o pausa
Byte 5	Valor de la nota 0
Byte 6	Valor de la nota 1
Byte 7	Valor de la nota 2
Byte 8	Valor de la nota 3
Byte 9	Valor de la nota 4
Byte 10	Valor de la nota 5
Byte 11	Valor de la nota 6
Byte 12	Valor de la nota 7
Byte 13	Número de notes
Byte 14	Usat per debuggar
Byte 15	Usat per debuggar

Taula 8.1: Continguts dels bytes que s'envien a cada lectura del DSP

8.1.5 EEPROM

La memòria EEPROM[30] (Electrically-Erasable Programmable Read-Only Memory) és un tipus de memòria ROM que pot ser programada, esborrada i reprogramada elèctricament. És una memòria no volàtil, és a dir, manté el seu estat quan se li deixa de subministrar energia elèctrica.

L'Arduino Mega 2560 disposa de 4KB de memòria EEPROM i també disposa de la llibreria EEPROM.h[31] per llegir-hi i escriure-hi.

Les dades que són necessàries guardar a la memòria EEPROM són: l'últim patró de notes que s'ha fet servir, és a dir, el nombre de notes d'aquest patró i els valors de les notes. Per escriure a la EEPROM es fa servir la funció EEPROM.write() que se li passa per paràmetre la dada i la direcció on es vol escriure.

Exemple:

```
EEPROM.write(numNotesAddress, numNotes);
```

A l'inici del programa, o quan es selecciona un patró, és necessari carregar les dades d'aquell patró, per tant es llegeix de la memòria EEPROM amb la funció EEPROM.read() que se li passa com a paràmetre l'adreça d'on es vol llegir.

Exemple:

```
byte numNotes = EEPROM.read(numNotesAddress);
```

Tant el valor de lectura com el d'escriptura d'aquestes funcions són d'1 byte, tot i que n'existeixen d'altres per a guardar altres tipus de variables.

8.2 C5515 eZdsp

La programació del DSP ha estat sens dubte la part més complicada i feixuga del projecte. Tot i la força documentació que proporciona Texas Instrument per a l'ús i configuració del C5515 USB Stick, les instruccions solen ser confuses i difícils de seguir. Tot i així s'ha pogut programar el DSP per a que realitzi la seva funció: reproduir notes musicals i aplicar efectes de so (eco i reverberació).

8.2.1 Configuració del C5515

Les funcions d'inicialització i configuració del C5515 USB Stick són les següents:

- `USBSTK5515_init();`

Inicia els rellotges dels perifèrics de la placa. Es troba al fitxer `usbstk5515.c`.

- `PLL_config();`

Configura el PLL del processador a 120 MHz. En anteriors proves del programa es feia servir la funció `pll_frequency_setup()` del fitxer `PLL.c`, però donava errors com es veurà més endavant.

- `aic3204_hardware_init();`

Aquesta funció inicialitza els ports que fa servir el processador per comunicar-se amb el còdec AIC3204 com també el bus I2C. Es troba al fitxer `aic3204.c`.

- `aic3204_init();`

Aquesta funció inicialitza i configura el còdec AIC3204. Es troba al fitxer `aic3204.init.c`.

- `set_sampling_frequency_and_gain();`

Se li passa com a paràmetre la freqüència de mostreig i el guany de l'àudio a que es vol que funcioni el còdec. Se li passa el valor de 48000 Hz i un guany de 0 dB ja que es fan

servir altaveus. La funció configura el còdec AIC3204 amb aquests paràmetres. Es troba al fitxer `aic3204.init.c`.

Per últim cal configurar el EBSR (External Bus Selection Register)[23] de la següent forma:

- `SYS_EXBUSSEL = 0x6100;`

Si es volen fer servir els LED's o la pantalla OLED de la placa s'han de fer servir les següents funcions:

- `USBSTK5515_ULED_init();`

Inicialitza els quatre LED's programables. Es troba al fitxer `usbstk5515_led.c`.

- `oled_init();`

Inicialitza la pantalla OLED. Es troba al fitxer `oled_init.c`

- `SAR_init();`

Inicialitza els canals AD. Es troba al fitxer `pushbuttons5515.c`

- `echo_array_clear();`

`reverberation_array_clear();`

Aquestes dues funcions inicialitzen a 0 els vectors que es fan servir per a guardar les mostres d'àudio de la reverberació i de l'eco.

8.2.2 Estructura

L'estructura del programa del DSP és la següent (en pseudocodi):

```
//Espera activa a que l'Arduino estigui disponible
while(llegir_del_arduino() != 0) { }

//Bucle infinit
while(1){
    per i=0 fins i=nombre_de_notes
    {
        llegir_del_arduino();
        actualització_variables();
        generar_nota(i);
    }
}
```

Per a garantir la sincronia entre l'Arduino i el DSP, el C5515 primerament s'espera a que l'Arduino estigui disponible, per això es fa un espera activa i fins que la funció `llegir_del_arduino()` no retorna 0 (que és quan significa que s'han llegit dades vàlides de l'Arduino) no s'inicia la seqüenciació de notes.

A cada iteració, s'aprofita entre nota i nota per fer una lectura a l'Arduino, això garanteix que els canvis que es produeixen, com el gir dels potenciòmetres o un canvi de patró o un canvi de nota a l'Arduino, es reflecteixi ràpidament en el DSP i doni l'efecte de que s'està produint en temps real.

8.2.3 Generació d'àudio

La generació del so es fa a la funció `generateNote()`, se li passa com a paràmetre la freqüència, l'amplitud i el temps de la nota.

L'àudio és una ona sinusoidal que es genera amb la funció `sine()` de la llibreria `DSPLib.h`. Aquesta llibreria es proporciona per Texas Instruments i les funcions que conté estan optimitzades per al dispositiu. Cada cop que es crida la funció genera el valor de la sinusoide corresponent.

Un cop es té el valor calculat es crida la funció `aic3204_codec_write()`. Aquesta funció escriu al canal dret i esquerra, passats com a paràmetre, del port d'àudio del còdec. Es pot trobar al fitxer `aic3204.c`.

El codi queda així:

```
per i=0 fins i==temps_de_la_nota
{
    sortida = generar_sinus();
    sortida_dret = sortida;
    sortida_esquerra = sortida;
    aic3204_codec_write(sortida_dret, sortida_esquerra);
}
```

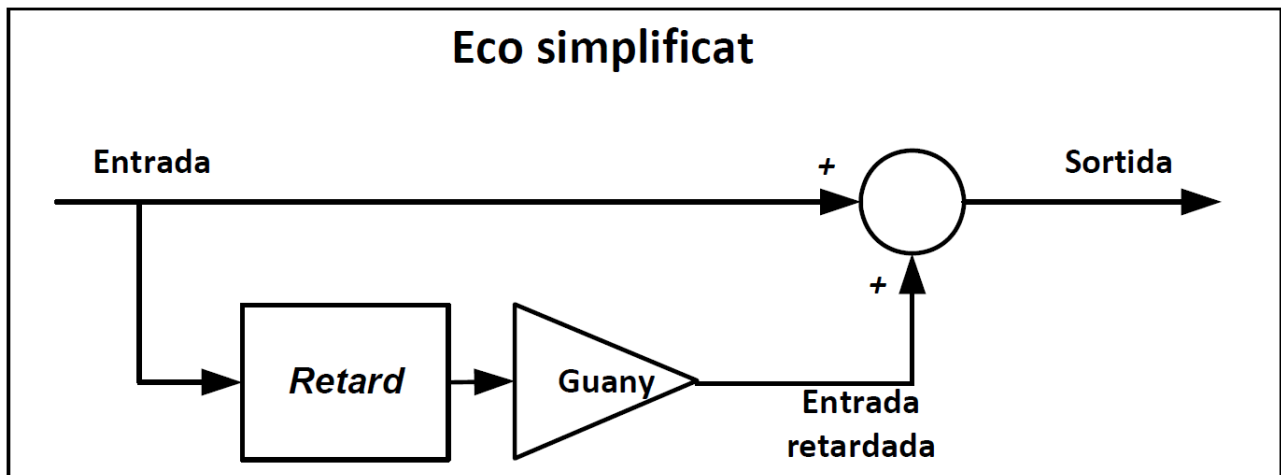
Si hi ha activitat un efecte de so, es modifica la dada *sortida* abans de ser escrita al port d'àudio com es veu al pròxim punt.

8.2.4 Efectes de so

Els efectes de so que s'han implementat en el projecte són l'eco i la reverberació. Texas Instrument proporciona uns codis d'exemple però que l'efecte que aconseguix és bastant pobre així que han estat modificats lleugerament per a que es noti més l'efecte. Tot i així el resultat obtingut és millorable.

8.2.4.1 Eco

L'eco és l'efecte sonor que es produeix quan un so emès arriba a l'oient per més d'una via. Per simplificar-ho es suposa que només arriba l'eco per un camí. El següent diagrama mostra com funciona l'efecte del eco.

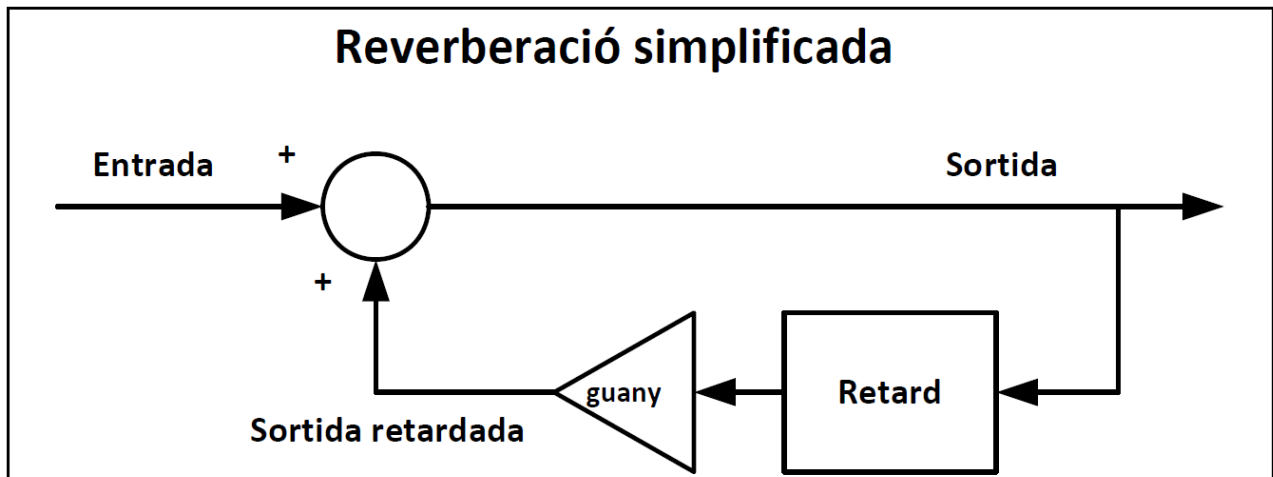


Il·lustració 8.3: Diagrama de l'eco per un camí

Per tant, en la pràctica es guarden les mostres d'àudio al vector i al cap d'un retard N se li sumen a la mostra d'àudio actual i s'aconsegueix l'efecte. En aquest eco el temps de retràs i el *feedback* són constants. Es pot trobar la funció `echo()` al fitxer `echo.c`.

8.2.4.2 Reverberació

La reverberació és semblant a l'eco però amb la diferència que es retarda la senyal de sortida i se li suma a l'entrada, és a dir, que és un efecte retroalimentat. Com en l'eco, només és té en compte un camí de retroalimentació.



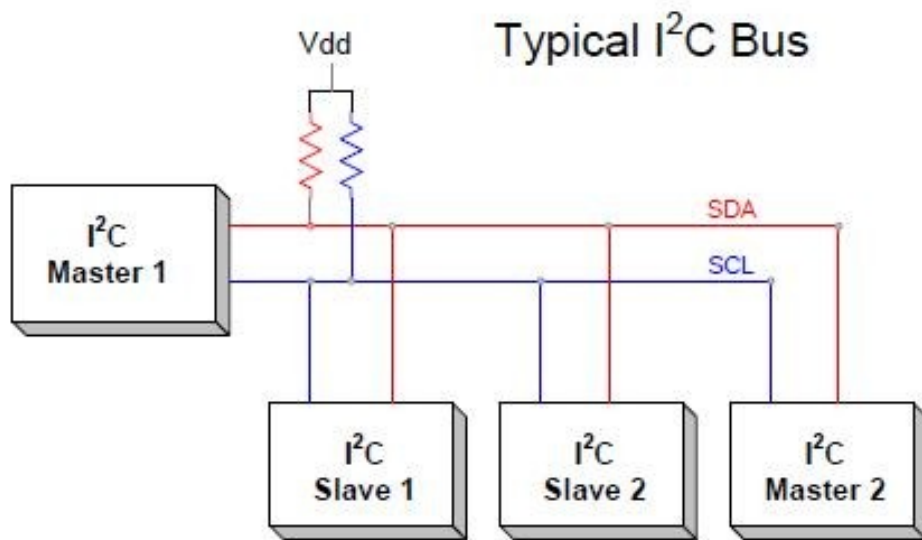
Il·lustració 8.4: Diagrama de la reverberació simplificada

En la pràctica s'agafa la mostra d'àudio actual, es manipula sumant-li la reverberació (sortida retardada) i a part d'escriure's al port d'àudio també es guarda al vector de reverberació per tornar a ser sumada més endavant. Es pot trobar la funció `reverberation()` al fitxer `reverberation.c`.

8.2.5 I2C

Per fer una petició al bus I2C s'utilitza la funció `USBSTK5515_I2C_read()` que se li passa com a paràmetre l'adreça del bus I2C, el punter al vector on es volen guardar les dades llegides i la quantitat de bytes que es volen llegir. Si es llegeixen dades vàlides, es guarden al vector i es retorna un 0; pel contrari, si arriba el *timeout* perquè no s'han rebut dades, aleshores la funció retorna -1 i es crida la funció de reinici del bus I2C.

En aquest projecte el C5515 USB Stick fa de mestre i l'Arduino d'esclau. Tant un com l'altre incorporen resistències pull-up internes per a la realització d'aquesta comunicació. També cal connectar els dos terres de les dues plaques.



Il·lustració 8.5: Esquema d'una connexió I2C (mestre i esclaus)

8.2.6 Escriptura del programa a la memòria NOR

Quan el C5515 USB Stick rep alimentació el bootloader busca imatges vàlides en el següent ordre: NOR Flash, NAND Flash, 16-bit SPI EEPROM, I2C EEPROM, MMC/SD i USB[32]. Per aquest projecte s'utilitzarà la memòria NOR.

Els passos a seguir per crear una imatge bootejable i escriure-la a la placa són els següents (és necessari usar el Code Compser Studio de Texas Instrument):

1. Primerament es compila el projecte, generant el fitxer TFG.out.
2. Es va al directori on s'ha generat el fitxer .out i s'executa la següent comanda (en windows):

```
hex55 -i TFG.out -o TFG_imatge.bin -boot -v5505 -b -serial8
```

Aquesta comanda crea la imatge binària.

3. Seguidament es carrega a la placa un petit programa proporcionat per Texas Instrument anomenat nor_writer.out.
4. A l'executar aquest programa, per la consola del CCS se'ns demana introduir la ruta de la imatge que volem escriure a la memòria NOR.
5. Si tot surt bé, per la mateixa consola ens indica els bytes escrits i si s'ha escrit la memòria correctament.
6. Ara es pot treure la corrent de la placa i tornar a donar-ne, de forma que bootejarà de la imatge escrita a la memòria NOR.

Han sortit alguns problemes donat a la falta de configuració d'alguns registres, com pot ser el EBSR. El EBSR mapeja els senyals del controlador del LCD, del I2S, UART, SPI i GPIO. Quan es debugava el programa des de l'ordinador no calia configurar-lo ja que l'emulador ja feia aquesta funció, però si es carrega un programa a la placa s'ha de configurar de la següent forma: `SYS_EXBUSSEL = 0x6100;`

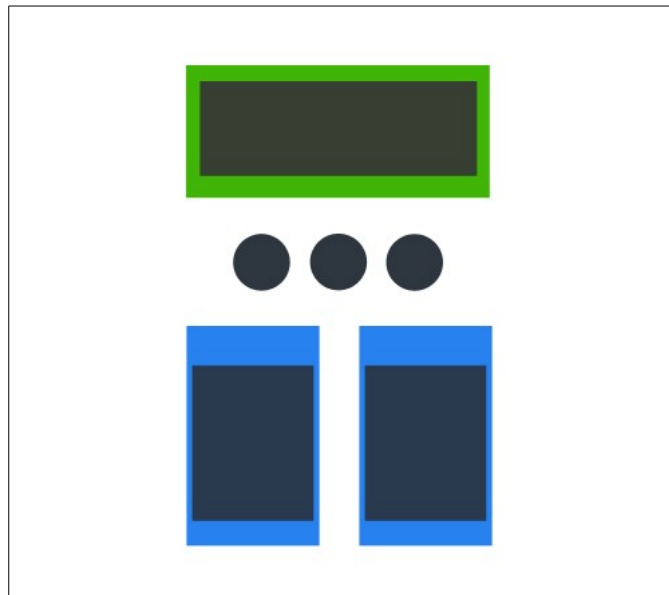
`SYS_EXBUSSEL` és un punter que apunta a l'adreça del EBSR (0x1C00). Configurant-ho així s'activa el mode desitjat per a que funcionin tots els perifèrics del processador.

9 Disseny i construcció de la caixa

9.1 Disseny

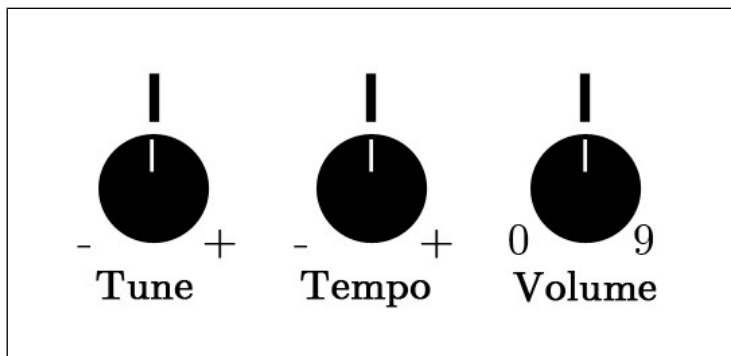
S'ha pensat en la distribució de la caixa de forma ergonòmica. Es disposen d'una pantalla LCD 16x2, tres potenciòmetres i dos teclats AD, per tant s'ha col·locat la pantalla a la part superior per a que quan es manipulin els potenciòmetres o es polsi els teclats no es tapi la pantalla amb les mans i no irrompi la seva visualització. Els potenciòmetres s'han posat al mig i els teclats a la part inferior.

La distribució es mostra en el següent esquema: pantalla (rectangle verd), tres potenciòmetres (circumferències) i dos teclats (rectangles blaus).

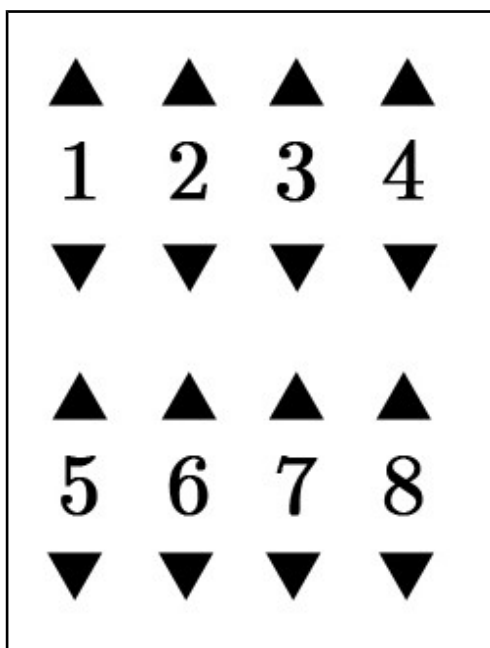


Il·lustració 9.1: Esquema de la distribució dels components

També s'han dissenyat uns adhesius per a que facin d'indicadors tal i com es mostra a les figures següents. A la pràctica s'han imprès en paper i han quedat adherits quan s'ha folrat la caixa.



Il·lustració 9.2: Indicadors dels potenciòmetres



Il·lustració 9.3: Indicador del teclat esquerre

▶	Save	Reset	▲ #Notes
Normal	Reverb	Echo	▼
Patterns 0	1	2	3
4	5	6	N/T

Il·lustració 9.4: Indicador del teclat dret

9.2 Resultat final de la caixa

A l'annexe s'adjunta el procés de construcció de la caixa. El resultat final és el següent.



Il·lustració 9.5: Caixa final del seqüenciador

10 Treball futur

En aquest projecte s'ha desenvolupat una primera versió d'un seqüenciador de notes musicals digital, però hi ha molts punts que es poden millorar que no s'han pogut realitzar per limitacions de temps i per falta de recursos i coneixements. A continuació es mostra un seguit de possibles millores a les quals pot ser sotmès el dispositiu:

- **Alimentació externa:** Una de les limitacions actual és que el sistema només pot ser alimentat per USB i això requereix un ordinador que l'alimenti. Una millora seria la possibilitat de ser connectat per la corrent alterna mitjançant un transformador de 220V alterns a 12V continus.
- **Pantalla LCD:** Com s'ha comprovat la pantalla actual queda petita per mostrar tota la informació necessària. Una mida més correcta podria ser una pantalla de 20x4, com el model LCD2004[33], que a més funciona amb I2C i això estalviaria cables.
- **MIDI:** La tecnologia MIDI (Musical Instrument Digital Interface) és un estàndard tècnic de comunicació entre equips musicals electrònics i fer que, per exemple, la seqüenciació sigui reproduïda en un altre instrument digital o el mateix ordinador. Aefgir una comunicació MIDI seria una millora molt important ja que és una tecnologia força usada en el món del instruments musicals electrònics.
- **Efectes de so:** En aquest projecte només s'ha implementat els efectes de l'eco i la reverberació però existeixen molts d'altres que també resulten interessants des del punt de vista musical. Alguns d'aquests són:
 1. Phaser: Consisteix en la modulació del senyal aconseguint un so metàl·lic.
 2. Flanger: Consisteix en mesclar la senyal original a una senyal idèntica però enrederida entre 5 ms i 20 ms. Això produeix un efecte similar al *phaser* també metàl·lic.
 3. Chorus: Consisteix en retardar la senyal original i mesclar-la amb una senyal idèntica modulada de manera que dóna la sensació que estant sonant varis instruments.
 4. Tremolo: Consisteix en modular l'amplitud de la senyal donant la sensació que el so 'tremola'.
 5. Distorsió: Consisteix en afegir soroll al senyal. Tot i que sembli poc lògic és

bastant usat en les produccions musicals per crear un efecte de baixa qualitat d'àudio del so de manera intencionada.

6. **Equalització:** Consisteix en la modificació del volum en certes freqüències. Es sol utilitzar un filtre pels baixos, un filtre pels mitjans, un filtre pels aguts i un filtre passa-baixos general (*cut-off*).
- **Efectes de so parametritzables:** Cada efecte de so té uns paràmetres que poden ser modificats mitjançant *knobs* o els polsadors. Per exemple, de l'eco es podria modificar el nombre de camins o el temps de retard del senyal.
 - **ADSR envelope:** L'ADSR (Attack Decay Sustain Release) són paràmetres d'una senyal que poden ser modificats per crear sons més interessants. L'*attack* és el temps que es tarda en assolir el primer pic en l'amplitud. El *decay* és el temps que tarda fins arribar al nivell del *sustain*. El *sustain* és el nivell en què el so és constant fins que deixa de sonar. El *release* és el temps que tarda el so d'anar del nivell del *sustain* fins a zero.
 - **PCB personalitzada:** Es pot dissenyar una placa PCB shield per a l'Arduino Mega que disposes de botons, potenciòmetre i la pantalla integrats. Això estalviaria molts cables, reduint soroll, i també molt d'espai de forma que la caixa podria ser més petita.
 - **Caixa personalitzada:** Es pot dissenyar una caixa a la mida adient per a estalviar espai i amb els forats necessaris per a que el muntatge sigui més pràctic i fàcil.
 - **Aplicació Android:** Es pot incorporar un dispositiu bluetooth en l'Arduino per poder comunicar-se amb un mòbil i amb una aplicació poder modificar els paràmetres del seqüenciador. Podria ser útil en cas de que per problemes d'espai físic no s'arribés al seqüenciador.

11 Conclusions

En general, valoro molt positivament la realització d'aquest projecte. He obtingut coneixements diversos, tant com del món de l'àudio digital o com de l'electrònica. A part també m'ha servit per consolidar, en major o menor mesura, els coneixements apresos en les assignatures d'aquest grau, sobretot de les assignatures de l'especialitat.

El principal objectiu del projecte era construir un seqüenciador de notes musicals digital mitjançant la utilització d'un Arduino i un DSP. Crec que aquest objectiu s'ha aconseguit satisfactòriament ja que el prototip construït funciona tal i com s'havia especificat. Tot i així considero que té moltes millores per davant i que el farien un producte competitiu de veritat. Aquestes millores requeririen més inversió en disseny i programació mantenint el hardware que s'ha utilitzat. És a dir, que s'encarien els costos de recursos humans i costos indirectes però mantenint els costos de hardware i software.

He pogut comprovar com el món de l'àudio digital és un món bastant complex on es requereixen coneixements molt avançats, tant de tecnologia com de síntesi musical digital. M'ha resultat molt interessant adonar-me de com ha influenciat la informàtica en els instruments musicals electrònics, ja siguin pedals de guitarra, sintetitzadors o taules de mescles.

Pel que fa a la programació de l'Arduino ha estat fàcil com s'esperava, ja que disposa de moltes llibreries tant oficials com fetes per la comunitat. També cal remarcar que no s'aprofita tot el potencial que ofereix l'Ardunio, per tant es podria substituir per un PIC amb característiques similars. Això estalviaria costos de hardware però incrementaria en recursos humans atès que programar un PIC requereix molt més temps i més inversió. Respecte el DSP, ha estat la part de programació més complicada ja que s'ha de programar en certs moments en baix nivell, afegint que la documentació proporcionada pel fabricant sol ser confusa i poc entenedora. Per altra banda, tampoc s'ha aprofitat tota la potència que ofereix aquest DSP; com ja s'ha comentat anteriorment, es poden afegir moltes millores com filtres de so i altres efectes de senyals molt interessants, per a realitzar aquestes implementacions caldria una major inversió en el temps de programació del DSP.

La creació de la caixa també ha estat un repte. Haver de fer els forats corresponents pels components no ha resultat ser una tasca difícil però tampoc senzilla, on ha calgut paciència i la utilització d'eines com trepants o serres. El resultat de la caixa ha estat molt bo, presenta un bon aspecte i una bona robustesa.

12 Bibliografia

- [1] Matt Friedman. *Roland Jupiter-8*. 2008. URL: <http://www.vintagesynth.com/roland/jup8.php>
- [2] Matt Friedman. *Roland Juno-60*. 2008. URL: <http://www.vintagesynth.com/roland/juno60.php>
- [3] Elektron. *Elektron Monomachine - Legacy Products*. 2004. URL: <https://www.elektron.se/legacy-products/>
- [4] Cyclone Analogic. *Cyclone Analogic Bass Bot TT-303*. 2012. URL: <http://www.cyclone-analogic.fr/en/34-bass-bot-tt-303-0701980493430.html>
- [5] Arduino. *Arduino MEGA 2560*. 2015. URL: <https://www.arduino.cc/en/Main/ArduinoBoardMega2560>
- [6] Texas Instruments. *C5515 eZDSP USB Stick Development Tool*. 2010. URL: <http://www.ti.com/tool/tmdx5515ezdsp>
- [7] Arduino. *Arduino 1.6.12*. 2016. URL: <https://www.arduino.cc/en/Main/Software>
- [8] Texas Instruments. *Code Composer Studio (CCS) Integrated Development Environment (IDE)*. 2014. URL: <http://www.ti.com/tool/ccstudio>
- [9] Google. *Google Drive*. 2012. URL: <https://www.google.es/intl/ca/drive/>
- [10] Xavier Albaladejo. *Qué es SCRUM*. 2009. URL: www.proyectosagiles.org/que-es-scrum
- [11] Christian Felber, La economía del bien común. Deusto S.A. Ediciones . 9788423412808. 2011
- [12] Wikipedia. *Sound*. [Consulta: Desembre 2016]. URL: <https://en.wikipedia.org/wiki/Sound>
- [13] Wikipedia. *Audio Frequency*. [Consulta: Desembre 2016]. URL: https://en.wikipedia.org/wiki/Audio_frequency
- [14] B. H. Suits. *Frequencies of Musical Notes*. 1998. URL: <http://www.phy.mtu.edu/~suits/notefreqs.html>
- [15] Wikipedia. *Amplitud*. [Consulta: Desembre 2016]. URL: <https://ca.wikipedia.org/wiki/Amplitud>
- [16] Wikipedia. *Decibel*. [Consulta: Desembre 2016]. URL: <https://ca.wikipedia.org/wiki/Decibel>
- [17] Wikipedia. *Digital Audio*. [Consulta: Desembre 2016]. URL: https://en.wikipedia.org/wiki/Digital_audio
- [18] Wikipedia. *Nyquist Sampling Theorem*. [Consulta: Desembre 2016]. URL: https://en.wikipedia.org/wiki/Nyquist%E2%80%93Shannon_sampling_theorem
- [19] Wikipedia. *Convertidor analògic-digital*. [Consulta: Desembre 2016]. URL: https://ca.wikipedia.org/wiki/Convertidor_anal%C3%B2gic-digital
- [20] Wikipedia. *Conversió digital-analògica*. [Consulta: Desembre 2016]. URL: https://ca.wikipedia.org/wiki/Conversi%C3%B3_digital-anal%C3%B2gica
- [21] Tom Williamson. *Designing Microcontroller Systems for Electrically Noisy Environments*. 1993. URL: <http://www.nj7p.info/Manuals/PDFs/Intel/210313-002.PDF>
- [22] Spectrum Digital, Inc. *TMS320C5515 eZdspTMUSB Stick: Technical Reference*. 2010. URL: http://support.spectrumdigital.com/boards/usbstk5515/reva/files/usbstk5515_TechRef_RevA.pdf
- [23] Texas Instruments. *TMS320C5515 Fixed-Point Digital Signal Processor*. 2010. URL: <http://www.ti.com/lit/ds/symlink/tms320c5515.pdf>

- [24] Texas Instruments. *TLV320AIC3204 Ultra Low Power Stereo Audio Codec*. 2008. URL: <http://www.ti.com/lit/ds/symlink/tlv320aic3204.pdf>
- [25] Digi-Key. *MEC1-130-02-S-D-A*. 2000. URL: <http://www.digikey.com/product-detail/en/samtec-inc/MEC1-130-02-S-D-A/SAM8118-ND/1109105>
- [26] Waveshare. *AD Keypad*. 2015. URL: http://www.waveshare.com/wiki/AD_Keypad
- [27] Arduino. *LiquidCrystal Library*. 2010. URL: <https://www.arduino.cc/en/Reference/LiquidCrystal>
- [28] The Lab Book PAGES. *Switch Debouncing*. 2010. URL: <http://www.labbookpages.co.uk/electronics/debounce.html>
- [29] Arduino. *Wire Library*. 2006. URL: <https://www.arduino.cc/en/Reference/Wire>
- [30] Wikipedia. *EEPROM*. [Consulta: Desembre 2016]. URL: <https://en.wikipedia.org/wiki/EEPROM>
- [31] Arduino. *EEPROM Library*. 2006. URL: <https://www.arduino.cc/en/Reference/EEPROM>
- [32] Texas Instruments. *C5515 Boot-Image Programmer*. 2014. URL: http://processors.wiki.ti.com/index.php/C5515_Boot-Image_Programmer
- [33] Sunfounder. *I2C LCD2004*. 2016. URL: https://www.sunfounder.com/wiki/index.php?title=I2C_LCD2004

13 Annexe

13.1 Índex de taules

Taula 4.1: Costos dels recursos humans per rols.....	21
Taula 4.2: Costos en recursos humans per tasques.....	21
Taula 4.3: Costos hardware indirecte.....	22
Taula 4.4: Costos hardware components.....	22
Taula 4.5: Costos software.....	23
Taula 4.6: Costos indirectes.....	23
Taula 4.7: Costos per imprevistos.....	24
Taula 4.8: Cost total.....	24
Taula 4.9: Cost final del projecte.....	25
Taula 5.1: Matriu de sostenibilitat.....	27
Taula 6.1: Freqüències de les notes musicals.....	29
Taula 7.1: Connexions de la pantalla LCD a l'Arduino.....	40
Taula 8.1: Continguts dels bytes que s'envien a cada lectura del DSP.....	50

13.2 Índex d'il·lustracions

Il·lustració 1.1: Roland Jupiter-8.....	5
Il·lustració 1.2: Roland Juno-60.....	6
Il·lustració 1.3: Elektron Monomachine.....	6
Il·lustració 1.4: Cyclone Analogic TT-303 BassBot.....	7
Il·lustració 2.1: Disseny del sistema.....	10
Il·lustració 2.2: Arduino Mega 2560 Original.....	11
Il·lustració 2.3: Arduino Mega 2560 Clone.....	11
Il·lustració 2.4: TMS320C5515 eZdsp USB Stick.....	12
Il·lustració 3.1: Diagrama de Gantt.....	20
Il·lustració 6.1: Ones de diferent freqüències.....	28
Il·lustració 6.2: Amplitud d'una ona sinusoïdal.....	30
Il·lustració 6.3: Mostreig d'una ona amb profunditat de 4 bits.....	31
Il·lustració 6.4: Ports analògics de l'Arduino Mega 2560.....	32
Il·lustració 6.5: Esquema d'un altaveu. 1: imant, 2: bobina, 3: suspensió, 4: diafragma.....	33
Il·lustració 6.6: Exemple de la pertorbació que pot generar un EMI en l'àudio.....	34
Il·lustració 7.1: Arduino Mega 2560 R3 CH340G (clon).....	36
Il·lustració 7.2: Connector micro USB Tipus B mascle.....	36
Il·lustració 7.3: Connexió micro USB tipus B.....	36
Il·lustració 7.4: Diagrama de blocs del C5515 eZdsp USB Stick.....	37
Il·lustració 7.5: Minijack mascle i minijack femella de 3.5 mm.....	38
Il·lustració 7.6: Adaptador Samtec MEC1-130-02-S-D-A.....	39
Il·lustració 7.7: Slot d'expansió del C5515 USB Stick.....	39
Il·lustració 7.8: Pantalla LCD 16x2 i les seves connexions.....	40
Il·lustració 7.9: AD Keypad.....	41
Il·lustració 7.10: Esquema del AD Keypad.....	41
Il·lustració 7.11: Knobs de plàstic.....	42
Il·lustració 7.12: Potenciòmetre de 10 k Ω	42
Il·lustració 7.13: HUB USB amb interruptor.....	42
Il·lustració 8.1: Visualització de la pantalla LCD.....	46
Il·lustració 8.2: Visualització per camps de la pantalla LCD.....	47
Il·lustració 8.3: Diagrama de l'eco per un camí.....	56
Il·lustració 8.4: Diagrama de la reverberació simplificada.....	57
Il·lustració 9.1: Esquema de la distribució dels components.....	60
Il·lustració 9.2: Indicadors dels potenciòmetres.....	61
Il·lustració 9.3: Indicador del teclat esquerre.....	61
Il·lustració 9.4: Indicador del teclat dret.....	61
Il·lustració 9.5: Caixa final del seqüenciador.....	62

13.3 Glossari d'abreviatures

AD: Analogic / Digital

ADSR: Attack Decay Sustain Release

CCS: Code Composer Studio

CD: Compact Disc

DAW: Digital Audio Workstation

DCO: Digital Controlled Oscillator

DSP: Digital Signal Processor

EBSR: External Bus Selection Register

EEPROM: Electrically Erasable Programmable Read-Only Memory

EMI: Electromagnetic Interference

FFT: Fast Fourier Transform

I2C: Inter-Integrated Circuit

ICSP: In-Circuit Serial Programming

LCD: Liquid-Crystal Display

LED: Light Emetting Diode

LFO: Low Frequency Oscillator

MIDI: Musical Instrument Digital Interface

MISO: Master In / Slave Out

MMC: MultiMedia Card

MOSI: Master Out / Slave In

NAND: Negative AND

NOR: Negative OR

OLED: Organic Light Emetting Diode

PCB: Printed Circuit Board

PLL: Phase-Locked Loop

PWM: Pulse Width Modulation

RAM: Random Access Memory

RISC: Reduced Instruction Set Computing

ROM: Read Only Memory

RX: Receiver

SCK: Serial Clock

SCL: Serial Clock Line

SD: Secure Digital

SDA: Serial Data Line

SPI: Serial Peripheral Interface

SS: Slave Select

TX: Transmitter

TWI: Two Wire Interface

UART: Universal Asynchronous Receiver/Transmitter

USB: Universal Serial Bus

VCA: Voltage Controlled Amplifier

VCF: Voltage Controlled Filter

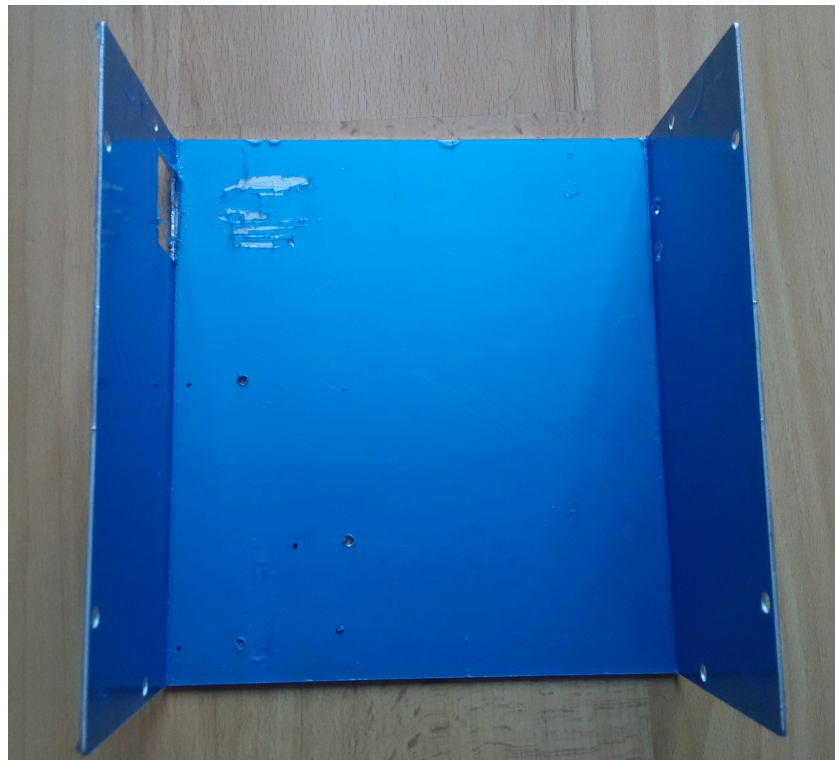
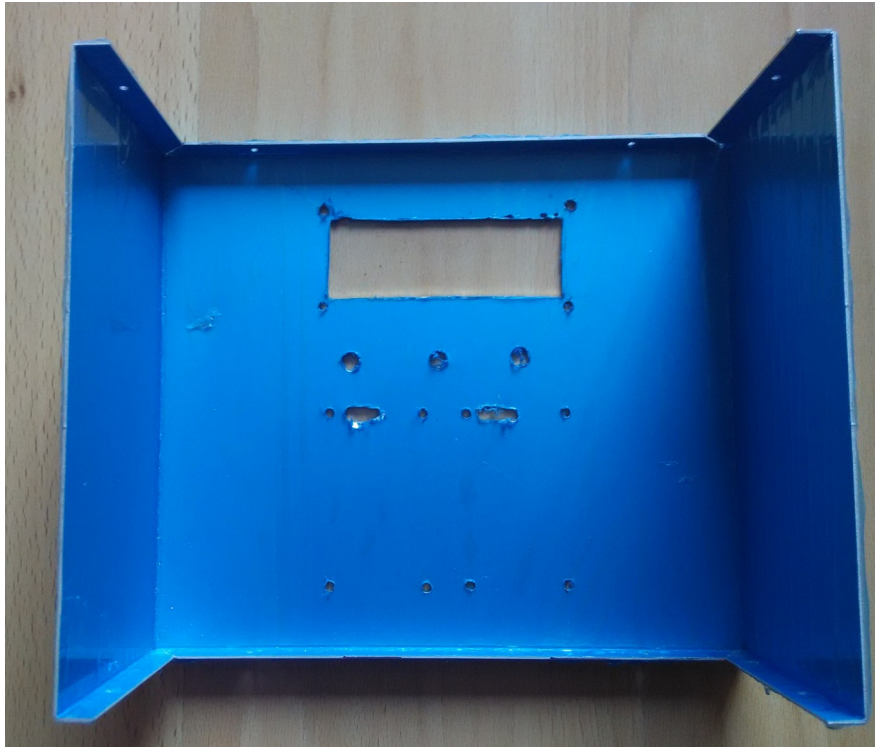
VCO: Voltage Controlled Oscillator

VSTi: Virtual Studio Technology Instrument

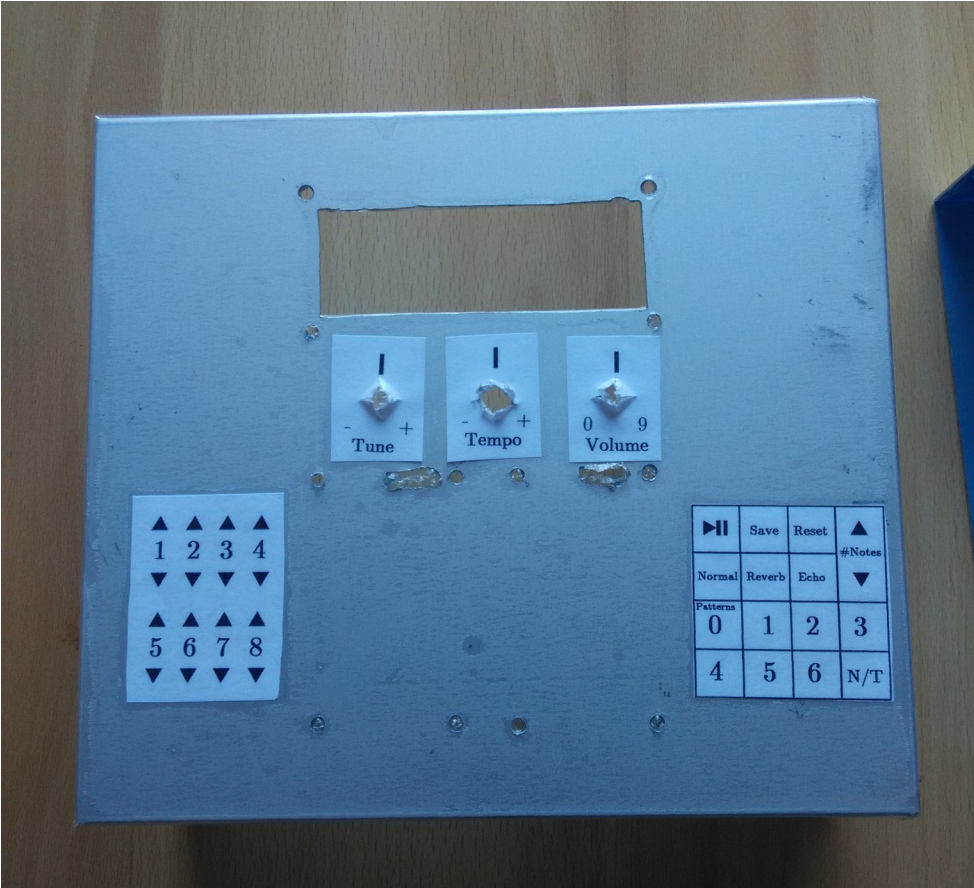
13.4 Construcció de la caixa

La caixa consta de dos estructures que s'ajunten amb un total de vuit cargols i porta un adhesiu protector de plàstic per defecte. Aquest adhesiu s'ha retirat per la part exterior però s'ha mantingut per la part interior.

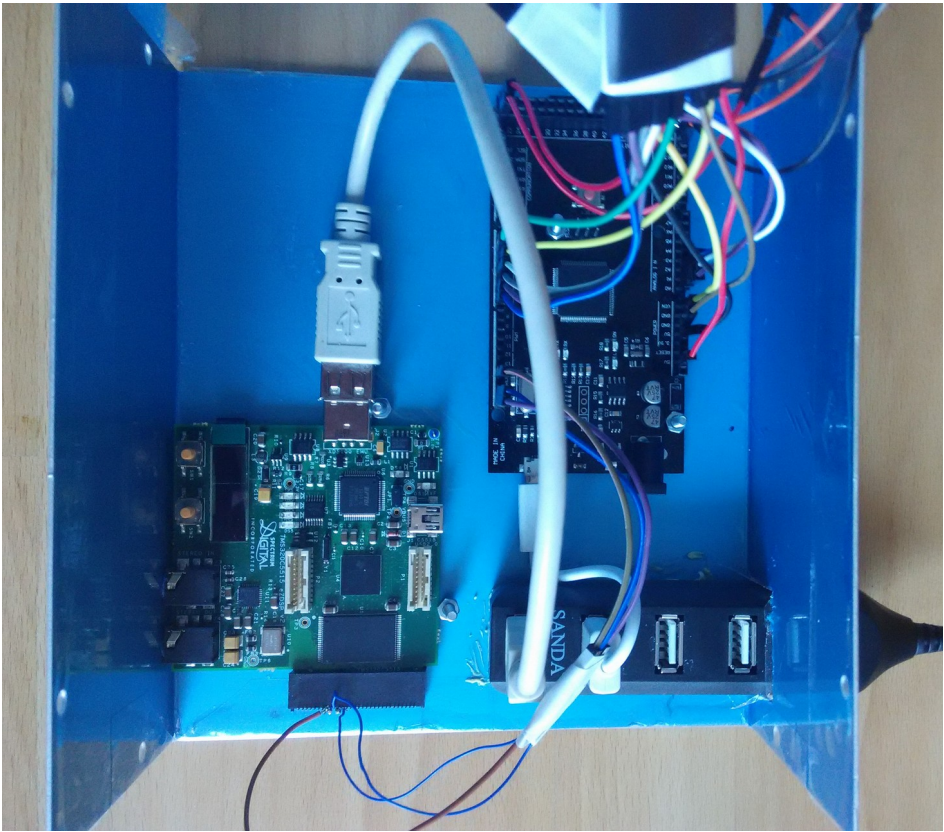
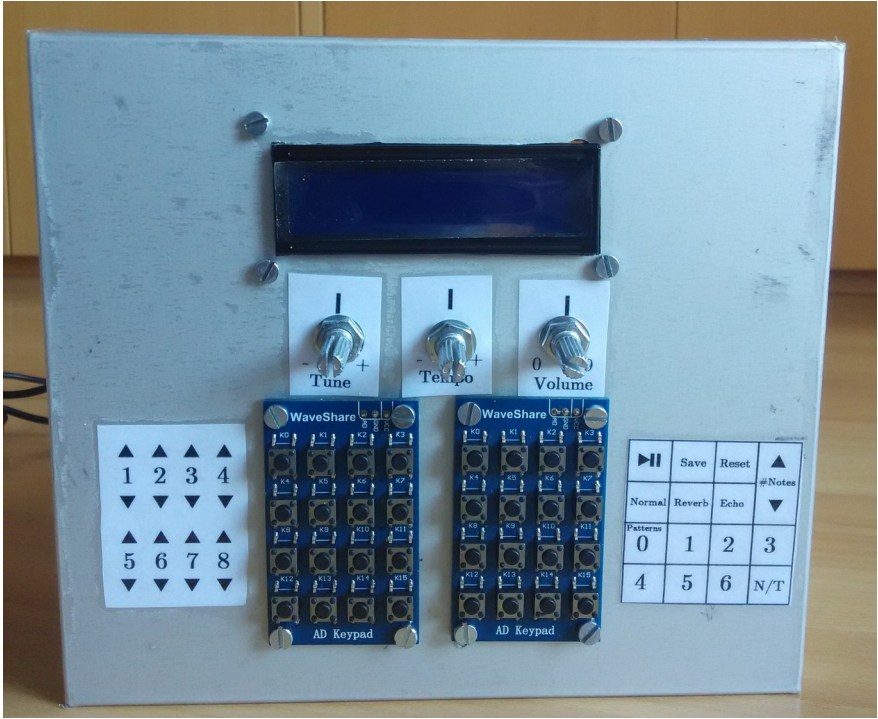
Forats:



Folratge i indicadors:



Incorporació dels components:



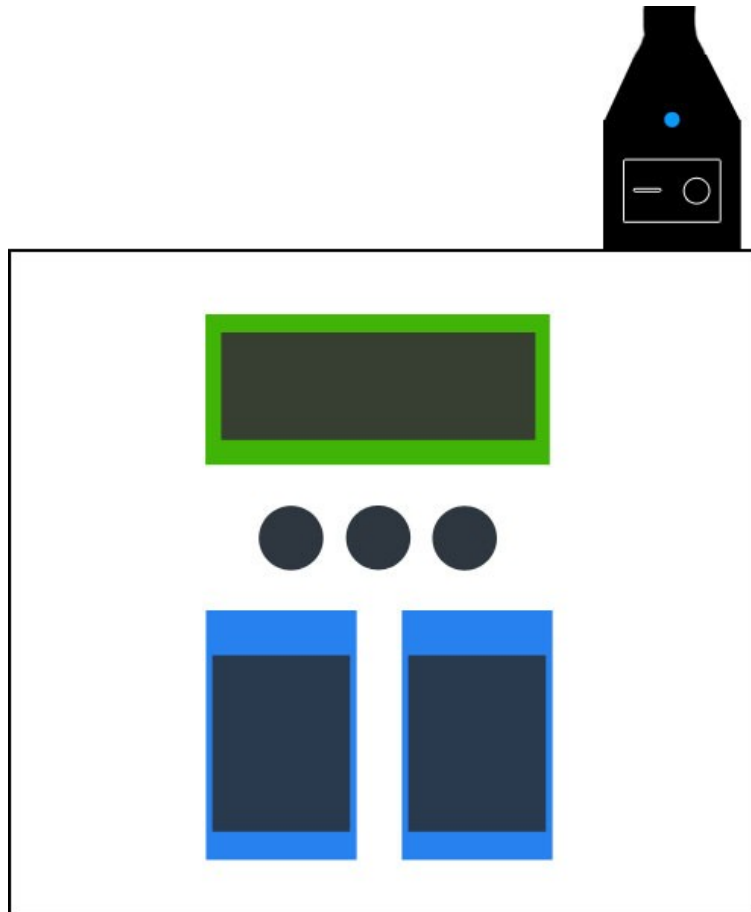
Aspecte final:



13.5 Manual d'usuari del seqüenciador

13.5.1 Encesa

El sistema s'encén i s'apaga amb un únic commutador situat a la part posterior de la caixa. Quan s'encén el commutador també s'encén un LED de color blau. Un cop s'encén el sistema ja apareix a la pantalla la informació del patró actual, que és l'últim que s'ha usat. A partir d'aquí ja es pot usar l'aparell.



13.5.2 Potenciòmetres

Els tres potenciòmetres del mig controlen el pitch (*tune*), el tempo i el volum de la seqüenciació, respectivament. Quan es situen a la meitat del gir (la línia blanca del *knob* apunta cap amunt) es considera que estan al punt neutre.

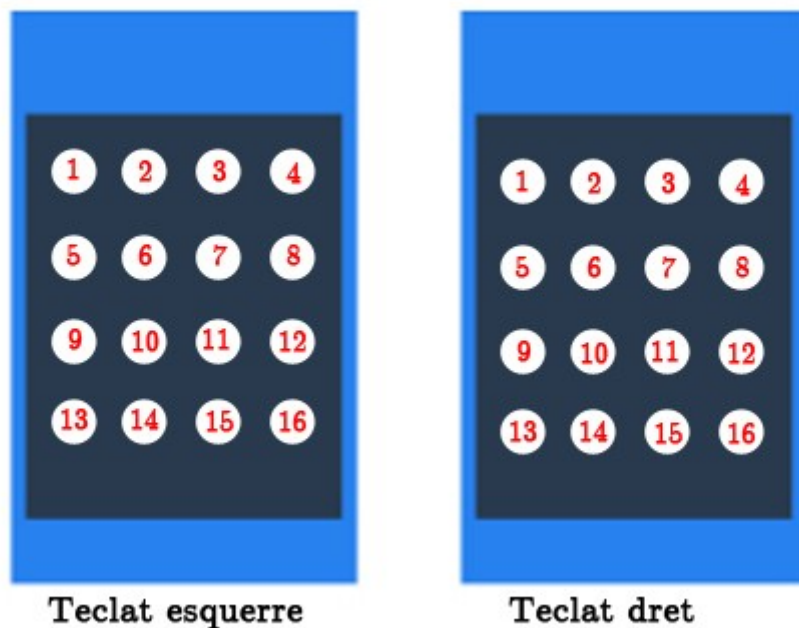
Potenciòmetre Tune: Si es gira cap a l'esquerra, el pitch aplicat és negatiu (so més greu). Si es gira cap a la dreta, el pitch aplicat és positiu (so més agut).

Potenciòmetre Tempo: Si es gira cap a l'esquerra, el tempo de la seqüència és més lent. Si es gira cap a la dreta, el tempo de la seqüència és més ràpid.

Potenciòmetre Volume: Estant al màxim girat a l'esquerra estaria al nivell 0 de volum, és a dir sense so. Estant al màxim girat a la dreta estaria al nivell 9, és a dir el nivell màxim de volum que ofereix el dispositiu.

13.5.3 Teclats

Els dos teclats serveixen per manipular la seqüenciació actual, canviar de patró, guardar o carregar un patró, activar efectes de so i canviar el nombre de notes de la seqüenciació. Tota la informació es mostra per la pantalla LCD tal i com està explicat al punt 8.1.1.



Teclat esquerre:

1	Incrementar nota/tempo de la nota de la posició 0
2	Incrementar nota/tempo de la nota de la posició 1
3	Incrementar nota/tempo de la nota de la posició 2
4	Incrementar nota/tempo de la nota de la posició 3
5	Reduir nota/tempo de la nota de la posició 0
6	Reduir nota/tempo de la nota de la posició 1
7	Reduir nota/tempo de la nota de la posició 2
8	Reduir nota/tempo de la nota de la posició 3
9	Incrementar nota/tempo de la nota de la posició 4
10	Incrementar nota/tempo de la nota de la posició 5
11	Incrementar nota/tempo de la nota de la posició 6
12	Incrementar nota/tempo de la nota de la posició 7
13	Reduir nota/tempo de la nota de la posició 4
14	Reduir nota/tempo de la nota de la posició 5
15	Reduir nota/tempo de la nota de la posició 6
16	Reduir nota/tempo de la nota de la posició 7

Teclat dret:

1	Permutar entre play i pausa
2	Guardar el patró actual a memòria interna
3	Reiniciar el patró actual a la nota C1 (sense guardar-lo a memòria)
4	Incrementar nombre de notes del patró
5	Desactivar efectes de so
6	Activar reverberació
7	Activar eco
8	Reduir nombre de notes del patró
9	Carregar el patró 0
10	Carregar el patró 1
11	Carregar el patró 2
12	Carregar el patró 3
13	Carregar el patró 4
14	Carregar el patró 5
15	Carregar el patró 6
16	Permutar entre el mode de modificació de nota o de tempo